Where Automation Connects.

MVI69E-GSC

CompactLogix™ or MicroLogix™ Platform

Generic ASCII Serial Communication Module

September 29, 2025

USER MANUAL

Your Feedback Please

We always want you to feel that you made the right decision to use our products. If you have suggestions, comments, compliments or complaints about our products, documentation, or support, please write or call us.

ProSoft Technology, Inc.
+1 (661) 716-5100
+1 (661) 716-5101 (Fax)
www.prosoft-technology.com
ps.support@belden.com

MVI69E-GSC User Manual
For Public Use.

September 29, 2025

Content Disclaimer

This documentation is not intended as a substitute for and is not to be used for determining suitability or reliability of these products for specific user applications. It is the duty of any such user or integrator to perform the appropriate and complete risk analysis, evaluation and testing of the products with respect to the relevant specific application or use thereof. Neither ProSoft Technology nor any of its affiliates or subsidiaries shall be responsible or liable for misuse of the information contained herein. Information in this document including illustrations, specifications and dimensions may contain technical inaccuracies or typographical errors. ProSoft Technology makes no warranty or representation as to its accuracy and assumes no liability for and reserves the right to correct such inaccuracies or errors at any time without notice. If you have any suggestions for improvements or amendments or have found errors in this publication, please notify us.

No part of this document may be reproduced in any form or by any means, electronic or mechanical, including photocopying, without express written permission of ProSoft Technology. All pertinent state, regional, and local safety regulations must be observed when installing and using this product. For reasons of safety and to help ensure compliance with documented system data, only the manufacturer should perform repairs to components. When devices are used for applications with technical safety requirements, the relevant instructions must be followed. Failure to use ProSoft Technology software or approved software with our hardware products may result in injury, harm, or improper operating results. Failure to observe this information can result in injury or equipment damage.

Printed documentation is available for purchase. Contact ProSoft Technology for pricing and availability.

For professional users in the European Union
If you wish to discard electrical and electronic equipment (EEE), please contact your dealer or supplier for further information.

Warning – Cancer and Reproductive Harm – www.P65Warnings.ca.gov

Agency Approvals & Certifications

Please visit our website: www.prosoft-technology.com

Contents

1 Start Here

To get the most benefit from this User Manual, you know the following skills:

- **Studio 5000 Logix Designer®:** launch the program, configure ladder logic, and transfer the ladder logic to the processor
- **Microsoft Windows:** install and launch programs, execute menu commands, navigate dialog boxes, and enter data
- **Hardware installation and wiring:** install the module, and safely connect and CompactLogix or MicroLogix devices to a power source and to the MVI69E-GSC module's application port(s)

1.1 System Requirements

The MVI69E-GSC module requires the following minimum hardware and software components:

- Rockwell Automation CompactLogix or MicroLogix 1500-LRP® processor (firmware version 10 or higher), with compatible power supply and one free slot in the rack, for the MVI69E-GSC module.

Important: The MVI69E-GSC module has a power supply distance rating of 4 (L43 and L45 installations on first 2 slots of 1769 bus). It consumes 500 mA at 5 VDC.

Important: For 1769-L23x processors, please make note of the following limitation:
1769-L23E-QBFC1B = 450 mA at 5 VDC (No MVI69E module can be used with this processor.)

- The module requires 500 mA of available 5 VDC power
- Rockwell Automation Studio 5000 Logix Designer version 16 or higher
- Rockwell Automation RSLinx® communication software version 2.51 or higher
- ProSoft Configuration Builder (PCB) (included)
- ProSoft Discovery Service (PDS) (included in PCB)
- Supported operating systems:
  - Microsoft Windows® 10 Professional
  - Microsoft Windows® 7 Professional
  - Microsoft Windows® XP Professional
- 128 Mbytes of RAM minimum, 256 Mbytes of RAM recommended
- 100 Mbytes of free hard disk space (or more based on application requirements)

Note: The Hardware and Operating System requirements in this list are the minimum recommended to install and run software provided by ProSoft Technology®. Other third-party applications may have different minimum requirements.

1.2 Package Contents

The following components are included with your MVI69E-GSC module, and are all required for installation and configuration.

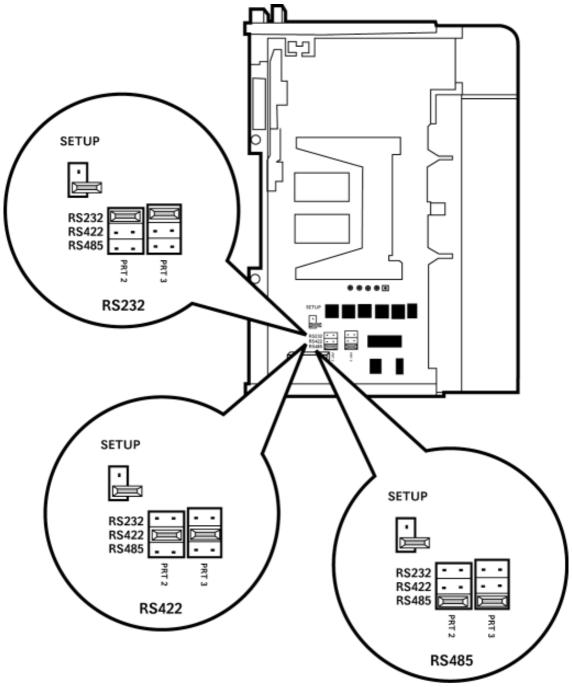
Important: Before beginning the installation, please verify that all the following items are present.

Qty.	Part Name	Part Number	Part Description
1	MVI69E-GSC Module	MVI69E-GSC	Generic ASCII Serial Communication Module
2	Adapter Cable	Cable #14	RJ45 to DB9 Male Adapter cable. For DB9 connection to module's serial application ports
2	Screw Terminal Adapter	1454-9F	DB9 female to 9-pin screw terminal. Used for RS422 or RS485 connections to Port 1 and 2 of the module

If any of these components are missing, please contact ProSoft Technology Support for replacement parts.

1.3 Setting Jumpers

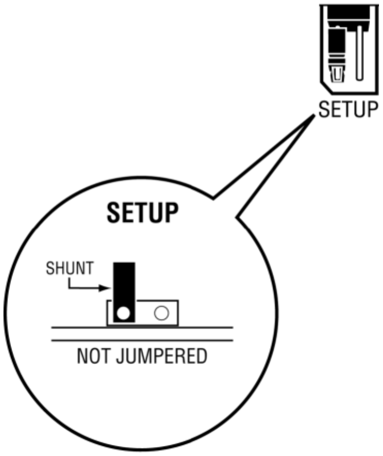
When the module is manufactured, the port selection jumpers are set to RS-232. To use RS-422 or RS-485, you must set the jumpers to the correct position. The following diagram describes the jumper settings.



Note: Jumper pin placement on the circuit board may vary.

The Setup Jumper acts as "write protection" for the module’s firmware. In "write protected" mode, the Setup pins are not connected, and the module’s firmware cannot be overwritten. The module is shipped with the Setup jumper OFF. If an update of the firmware is needed, apply the Setup jumper to both pins.

The following illustration shows the MVI69E-GSC jumper configuration, with the Setup Jumper OFF.



1.4 Install the Module in the Rack

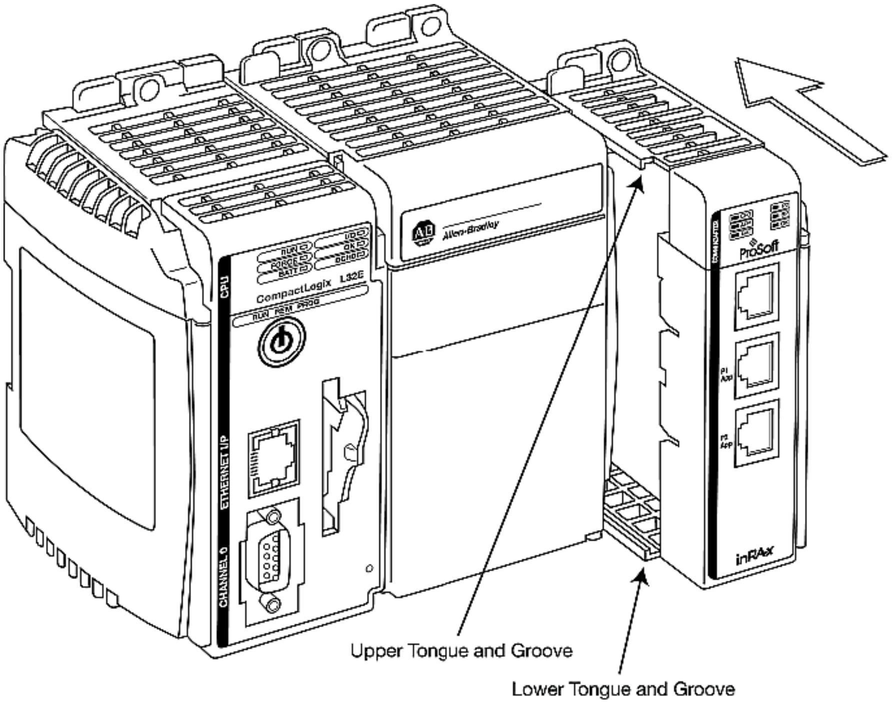
Make sure the processor and power supply are installed and configured before installing the MVI69E-GSC module. Refer to the Rockwell Automation product documentation for installation instructions.

Warning: Please follow all safety instructions when installing this or any other electronic devices. Failure to follow safety procedures could result in damage to hardware or data, or even serious injury or death to personnel. Refer to the documentation for each device to be connected to verify that suitable safety procedures are in place before installing or servicing the device.

After you verify the jumper placements, insert the MVI69E-GSC into the rack. Use the same technique recommended by Rockwell Automation to remove and install CompactLogix or MicroLogix 1500-LRP modules.

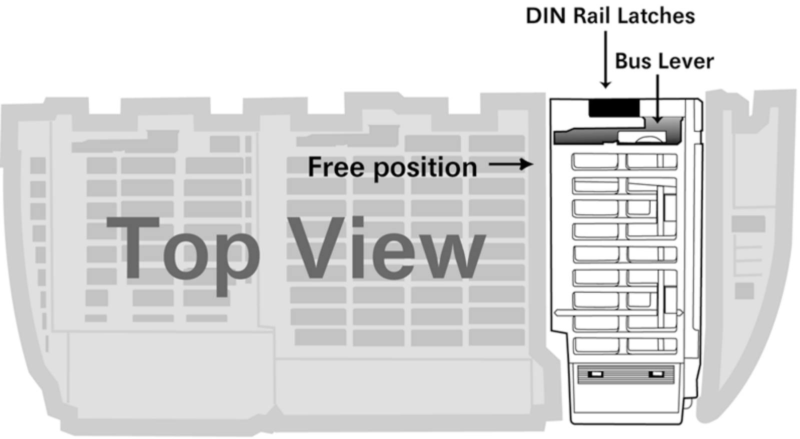
Warning: This module is not hot-swappable! Always remove power from the rack before inserting or removing this module, or damage may result in the module, the processor, or other connected devices.

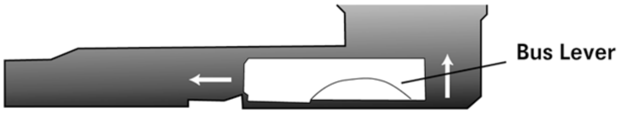
1 Align the module using the upper and lower tongue-and-groove slots with the adjacent module and slide forward in the direction of the arrow.



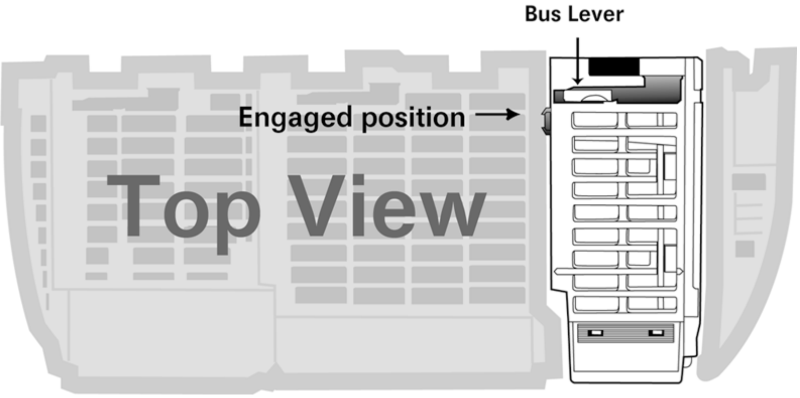
2 Move the module back along the tongue-and-groove slots until the bus connectors on the module and the adjacent module line up with each other.

3 Push the module’s bus lever back slightly to clear the positioning tab and move it firmly to the left until it clicks. Ensure that it is locked firmly in place.



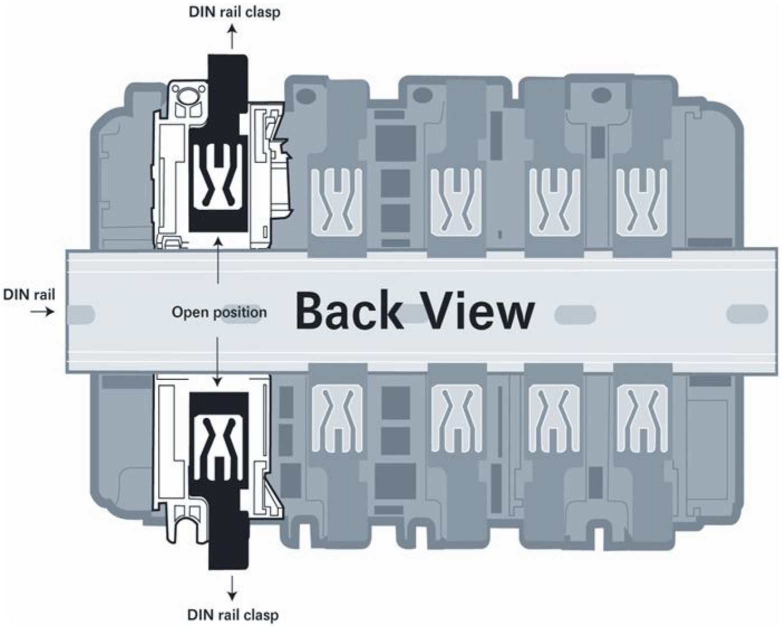


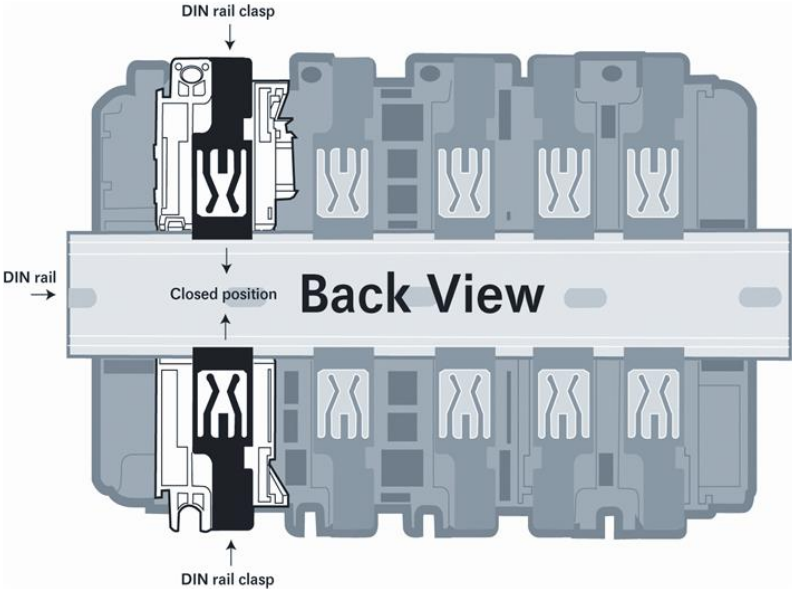
Move the Bus Lever to the left until it clicks



4 Close all DIN-rail latches.

5 Press the DIN-rail mounting area of the controller against the DIN-rail. The latches will momentarily open and lock into place.





2 Configuring the Module in RSLogix

To add the MVI69E-GSC module in Studio 5000, you must:

1 Create a new project in Studio 5000.

2 Add the module to the Studio 5000 project.

- You can manually create the module using a generic 1769 profile, and then manually configure the module parameters.

3 Download the *MVI69EGSC_AddOn_Rung.L5X* file from www.prosoft-technology.com.

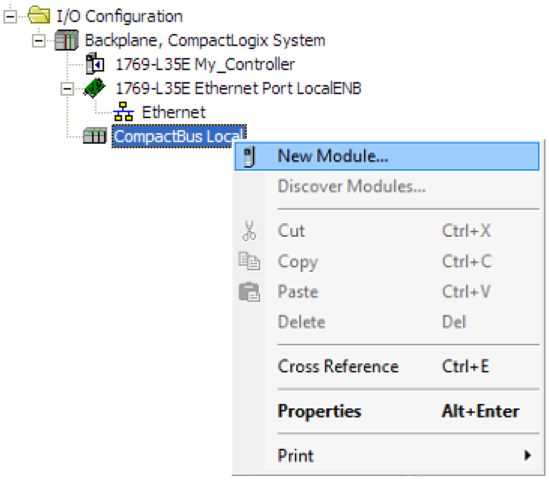
4 Import the Add-On Instruction (the .L5X file) into Studio 5000.

The .L5X file contains the Add-On Instruction, user-defined data types, controller tags and ladder logic required to configure the MVI69E-GSC module.

2.1 Creating the Module in a Studio 5000 Project

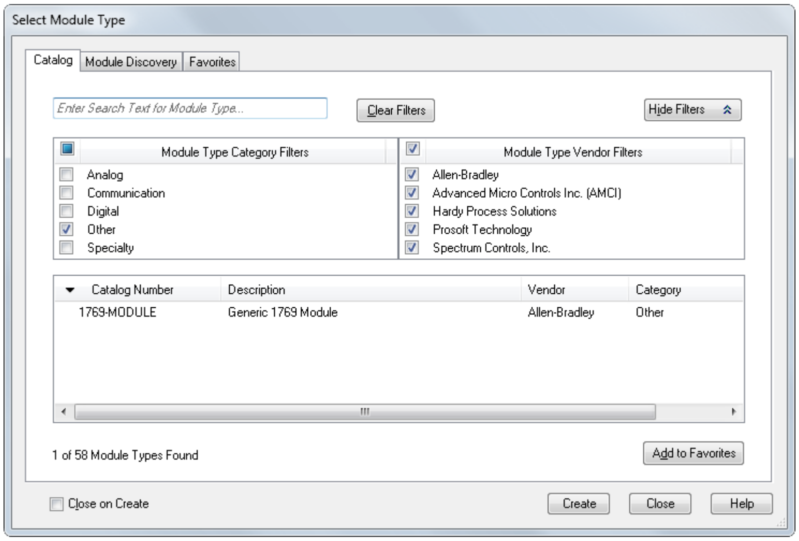
In a Studio 5000 project, you can manually create and configure the module using a generic 1769 profile.

1 Expand the **I/O CONFIGURATION** folder in the Project tree. Right-click the appropriate communications bus and choose **NEW MODULE**.



This opens the *Select Module Type* dialog box.

2 In the *Select Module Type* dialog, select the **1769-MODULE** and click on the **CREATE** button.



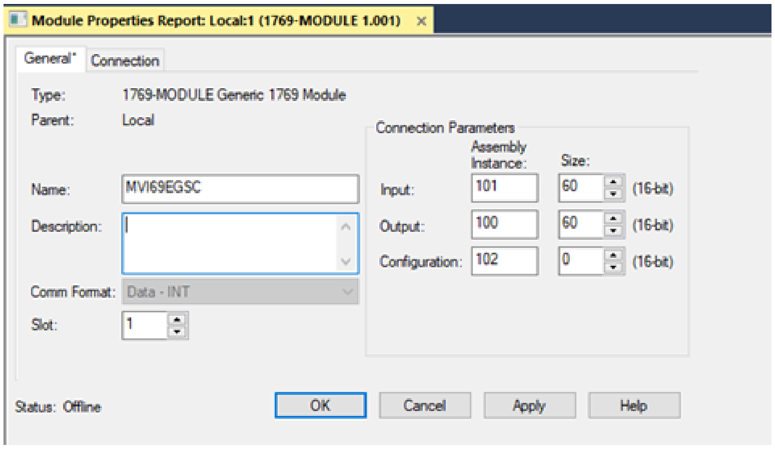
Module Type Category Filters	Module Type Vendor Filters
□ Analog	☑ Allen-Bradley
□ Communication	☑ Advanced Micro Controls Inc. (AMCI)
□ Digital	☑ Hardy Process Solutions
☑ Other	☑ Prosoft Technology
□ Specialty	☑ Spectrum Controls, Inc.

Catalog Number	Description	Vendor	Category
1769-MODULE	Generic 1769 Module	Allen-Bradley	Other

1 of 58 Module Types Found

3 Set the *Module Properties* values as follows:

Parameter	Value
Name	Enter a module identification string. Example: MVI69EGSC
Description	Enter a description for the module. Example: ProSoft communication module for Serial Modbus communications.
Comm Format	Select **DATA-INT**
Slot	Enter the slot number in the rack where the MV69E-GSC module is installed.
Input Assembly Instance	101
Input Size	60
Output Assembly Instance	100
Output Size	60
Configuration Assembly Instance	102
Configuration Size	0



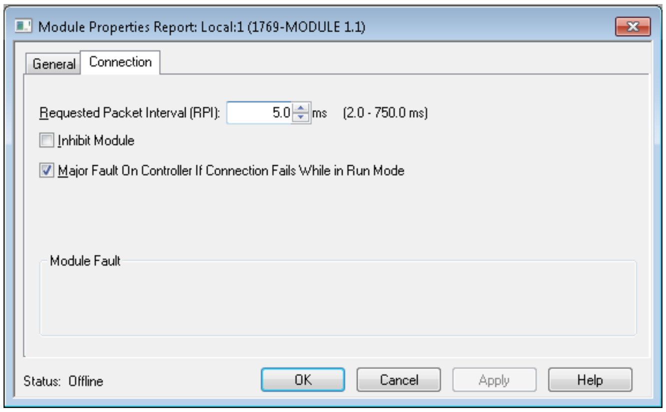
Type: 1769-MODULE Generic 1769 Module
Parent: Local
Name: MVI69EGSC
Description:
Comm Format: Data - INT
Slot: 1

Connection Parameters

	Assembly Instance:	Size:	
Input:	101	60	(16-bit)
Output:	100	60	(16-bit)
Configuration:	102	0	(16-bit)

Status: Offline

4 On the *Connection* tab, set the **REQUESTED PACKET INTERVAL** value for your project and click **OK**. A value of **10.0** ms or more is recommended.



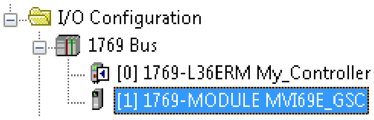
Requested Packet Interval (RPI): 5.0 ms (2.0 - 750.0 ms)

□ Inhibit Module

☑ Major Fault On Controller If Connection Fails While in Run Mode

Status: Offline

The MVI69E-GSC module is now visible in the *I/O Configuration* tree.



I/O Configuration
- 1769 Bus
  - [0] 1769-L36ERM My_Controller
  - [1] 1769-MODULE MVI69E_GSC

2.2 Importing the Add-On Instruction

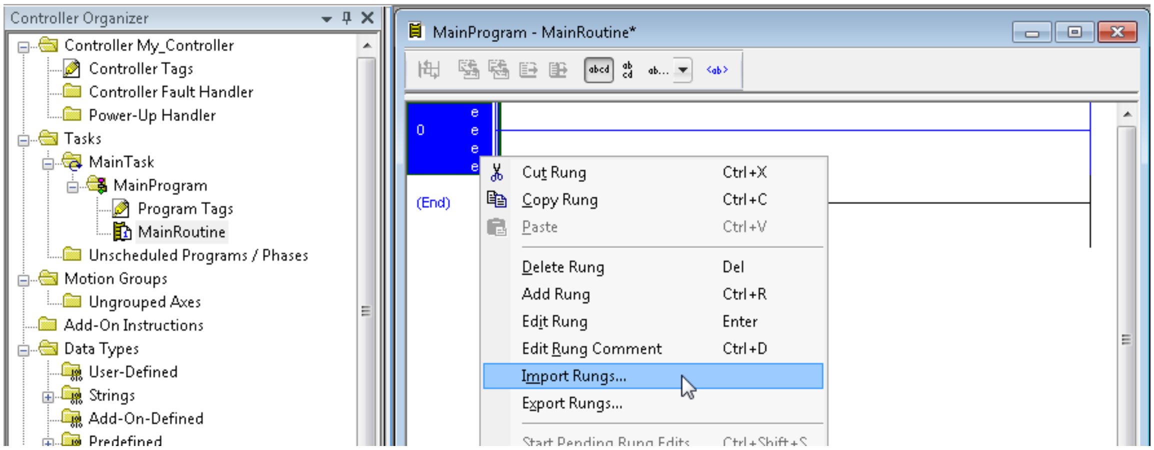
Note: This section only applies if you are using Studio 5000 version 16 or higher. If you are configuring the MVI69E-GSC module with an earlier version of Studio 5000, please refer to Adding the Module to an Existing CompactLogix Project (page 21).

The following file is required before you start this procedure. You can download it from www.prosoft-technology.com.

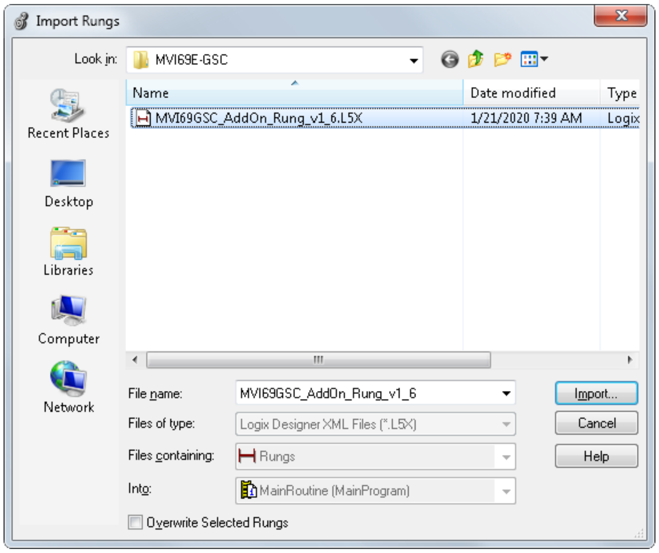
File Name	Description
MVI69EGSC_AddOn_Rung.L5X	File containing Add-On instruction, user defined data types, data objects and ladder logic required to set up the MVI69E-GSC module

2.2.1 Importing the Add-On Instruction

1. In RSLogix, expand the *Tasks* > *Main Task* > *Main Program* folder.
2. Double-click on the *MainRoutine* icon.
3. In an empty rung, right-click the mouse button to open a shortcut menu.
4. On the shortcut menu, choose **IMPORT RUNGS…**



5 Select the *MVI69EGSC_AddOn_Rung_v1_x.L5X* file. The Add-On Instruction file is located at www.prosoft-technology.com. Click on the **IMPORT** button.



Look in: MVI69E-GSC

Name	Date modified	Type
MVI69GSC_AddOn_Rung_v1_6.L5X	1/21/2020 7:39 AM	Logix

File name: MVI69GSC_AddOn_Rung_v1_6

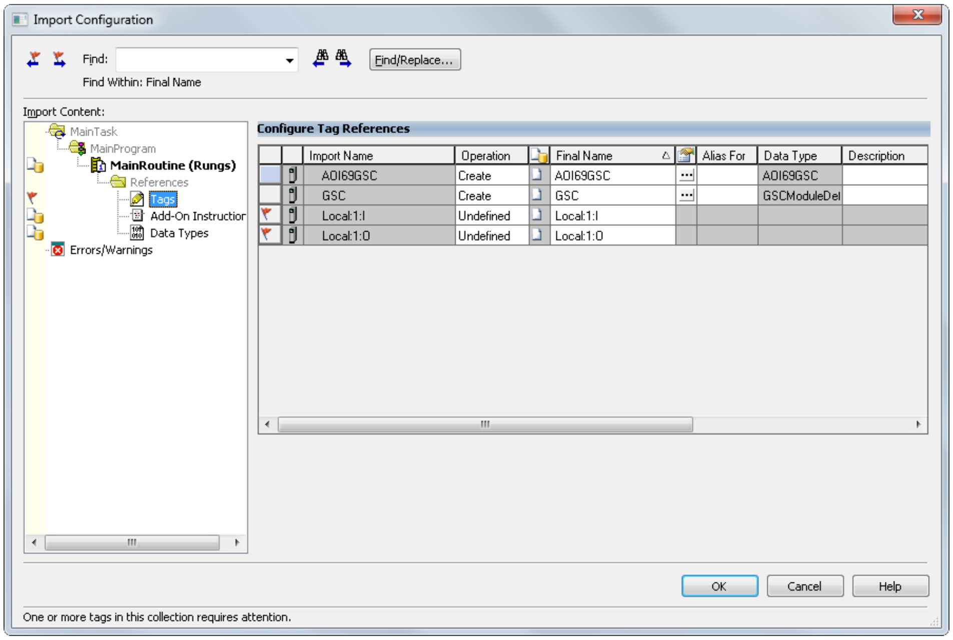
Files of type: Logix Designer XML Files (*.L5X)

Files containing: Rungs

Into: MainRoutine (MainProgram)

☐ Overwrite Selected Rungs

6 The following window displays the controller tags to be created during the import procedure. Make sure the slot number reference (Local:**x**) is correct.



Find:

Find Within: Final Name

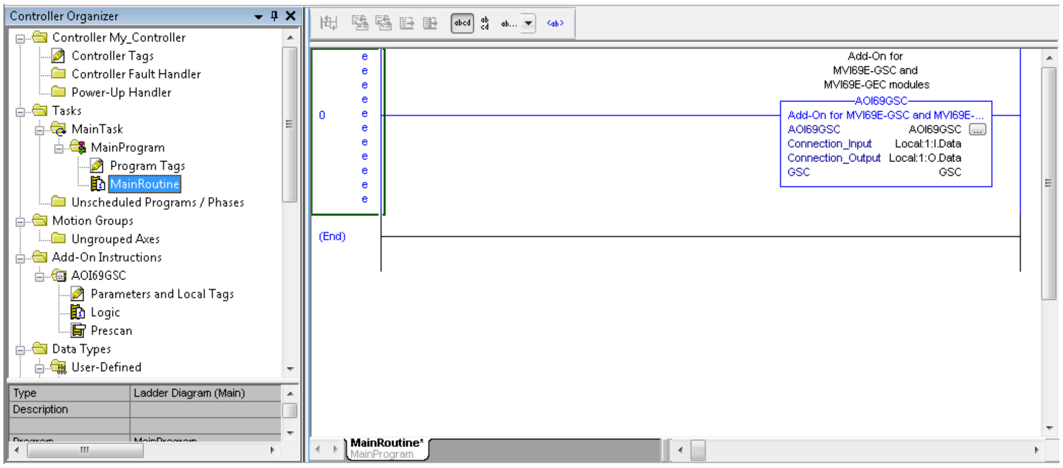
Import Content:

Configure Tag References

Import Name	Operation	Final Name	Alias For	Data Type	Description
AOI69GSC	Create	AOI69GSC		AOI69GSC	
GSC	Create	GSC		GSCModuleDef	
Local:1:I	Undefined	Local:1:I			
Local:1:O	Undefined	Local:1:O			

One or more tags in this collection requires attention.

7 When the import is complete, the new rung with the Add-On instruction is visible as shown in the following illustration.

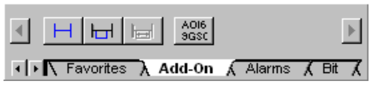


Add-On for MVI69E-GSC and MVI69E-GEC modules

AOI69GSC
Add-On for MVI69E-GSC and MVI69E-...

AOI69GSC	AOI69GSC
Connection_Input	Local:1:I.Data
Connection_Output	Local:1:O.Data
GSC	GSC

The procedure imports new user defined data types, data objects and the Add-On Instruction.

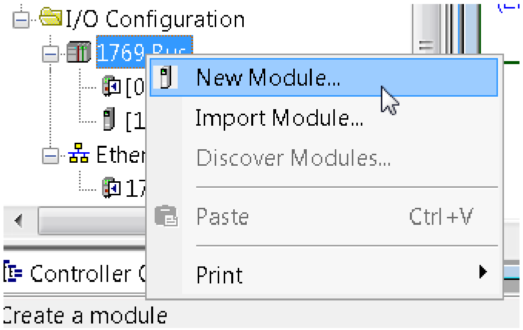


8 Save and download the project into the processor.

2.2.2 *Adding Multiple Modules in the Rack (Optional)*

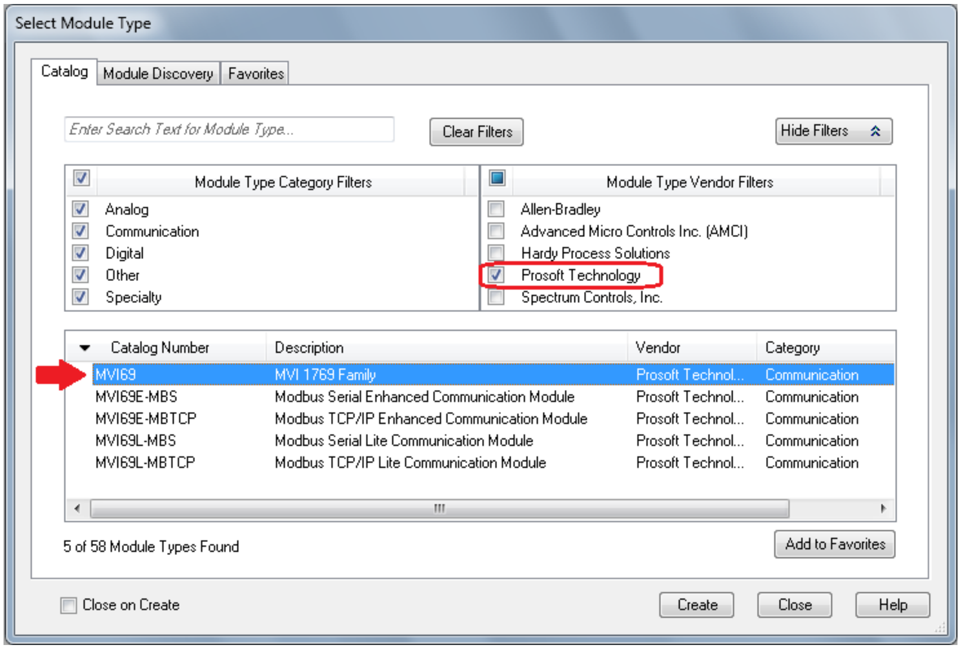
> **Important:** If your application requires more than one MVI69E-GSC module in the same project, follow the steps below.

1 In the *I/O Configuration* folder, right-click the mouse button to open a shortcut menu, and then select **NEW MODULE**.



2 Under the *Module Type Vendor Filters* section, select **ProSoft Technology**. Then select the **MVI69** option and click the **CREATE** button.

> **Note:** You can also add the module as a 'Generic 1769 module', as described in *Creating the Module in a Studio 5000 Project* on page 11.



☑ Module Type Category Filters

- ☑ Analog
- ☑ Communication
- ☑ Digital
- ☑ Other
- ☑ Specialty

■ Module Type Vendor Filters

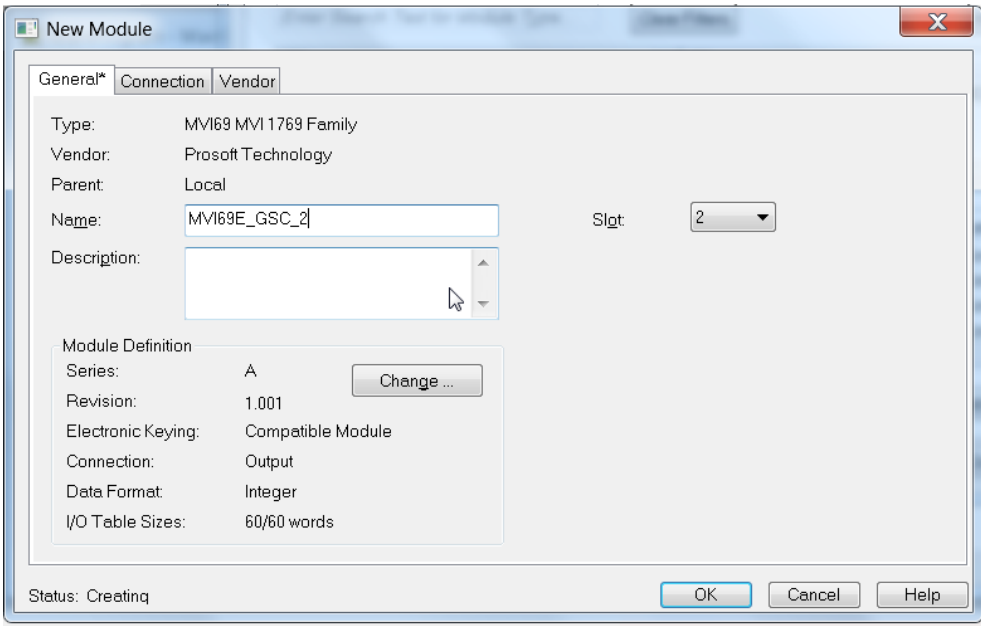
- □ Allen-Bradley
- □ Advanced Micro Controls Inc. (AMCI)
- □ Hardy Process Solutions
- ☑ Prosoft Technology
- □ Spectrum Controls, Inc.

Catalog Number	Description	Vendor	Category
MVI69	MVI 1769 Family	Prosoft Technol...	Communication
MVI69E-MBS	Modbus Serial Enhanced Communication Module	Prosoft Technol...	Communication
MVI69E-MBTCP	Modbus TCP/IP Enhanced Communication Module	Prosoft Technol...	Communication
MVI69L-MBS	Modbus Serial Lite Communication Module	Prosoft Technol...	Communication
MVI69L-MBTCP	Modbus TCP/IP Lite Communication Module	Prosoft Technol...	Communication

5 of 58 Module Types Found

□ Close on Create

3 In the *New Module* dialog, enter the *Name*, *Description*, and *Slot* options for your application, then click on the *Connection* tab.

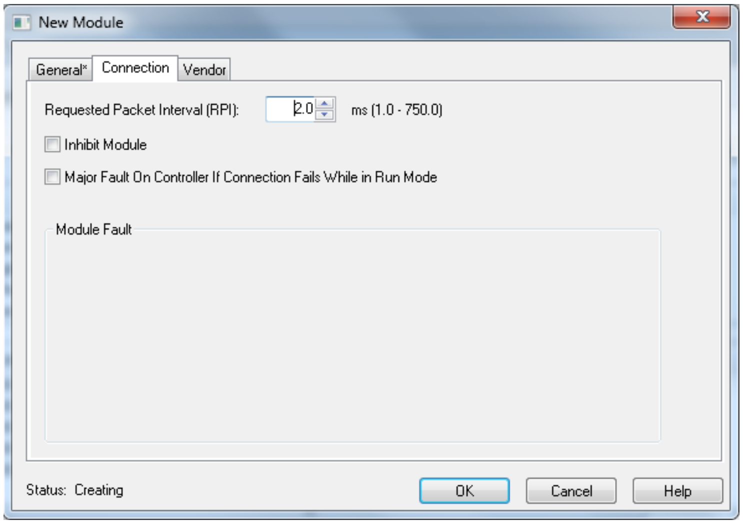


Type: MVI69 MVI 1769 Family
Vendor: Prosoft Technology
Parent: Local
Name: MVI69E_GSC_2
Slot: 2
Description:

Module Definition
Series: A
Revision: 1.001
Electronic Keying: Compatible Module
Connection: Output
Data Format: Integer
I/O Table Sizes: 60/60 words

Status: Creating

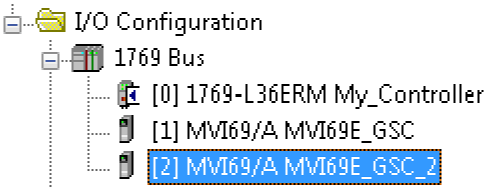
4 Select the *Request Packet Interval* value for scanning the I/O on the module. This value represents the minimum frequency the module will handle scheduled events. This value should not be set to less than 2 millisecond. Values between 2 and 10 milliseconds should work with most applications.



Requested Packet Interval (RPI): 2.0 ms (1.0 - 750.0)
□ Inhibit Module
□ Major Fault On Controller If Connection Fails While in Run Mode

Status: Creating

5 Click **OK** to confirm. The new module is now visible in the *I/O Configuration*:

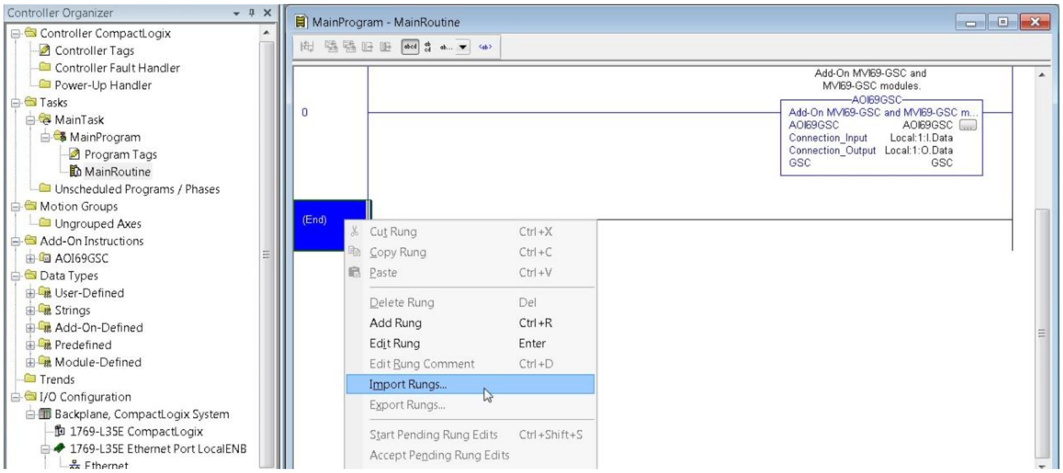


6 Expand the *Tasks* folder, and then expand the *MainTask* folder.

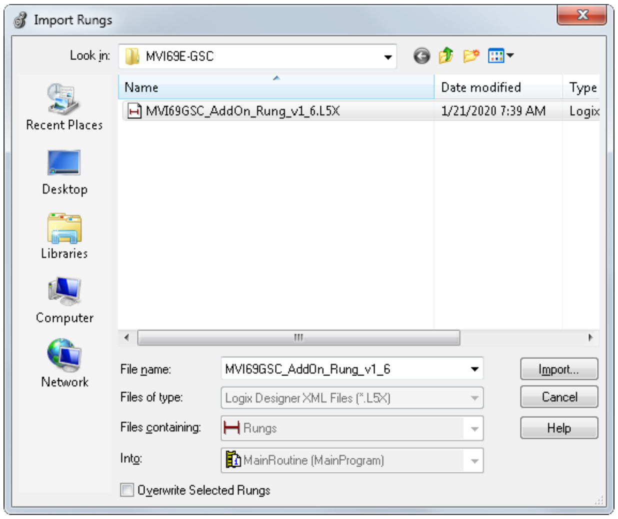
7 On the *MainProgram* folder, right-click the mouse button to open a shortcut menu. On the shortcut menu, choose **NEW ROUTINE**.

8 In the *New Routine* dialog box, enter the name and description of your routine, and then click **OK**.

9 Select an empty rung in the new routine, and then right-click the mouse button to open a shortcut menu. On the shortcut menu, choose **IMPORT RUNG…**



10 Select the *MVI69EGSC_AddOn_Rung.L5X* file, and click **IMPORT**.



MVI69GSC_AddOn_Rung_v1_6.L5X 1/21/2020 7:39 AM Logix

File name: MVI69GSC_AddOn_Rung_v1_6

Files of type: Logix Designer XML Files (*.L5X)

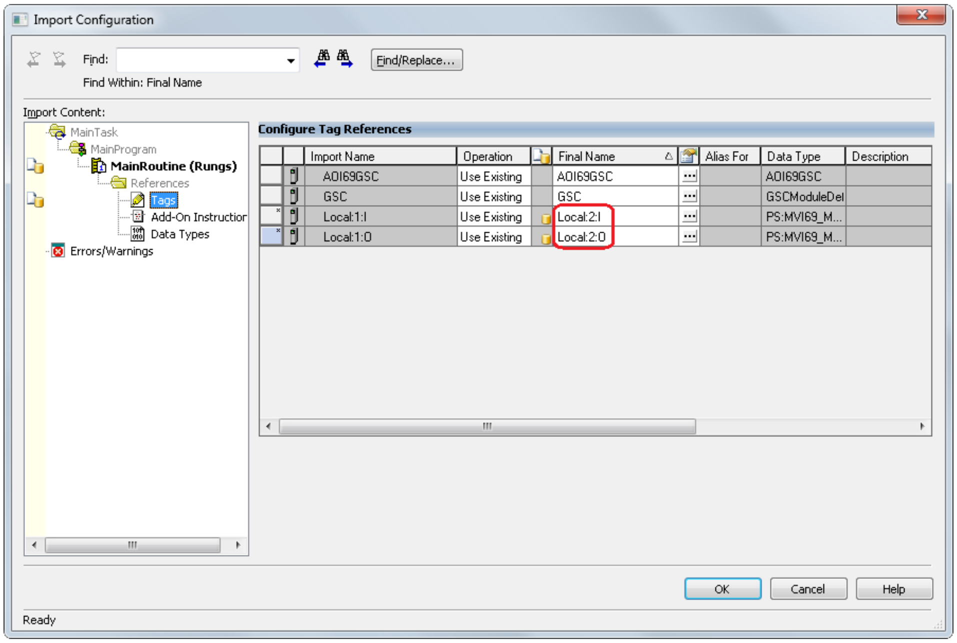
Files containing: Rungs

Into: MainRoutine (MainProgram)

☐ Overwrite Selected Rungs

11 This opens the **IMPORT CONFIGURATION** dialog box. Click the **TAGS** tab to show the controller tags in the AOI. You must edit the **FINAL NAME** column of the tags for the second module to make them unique.

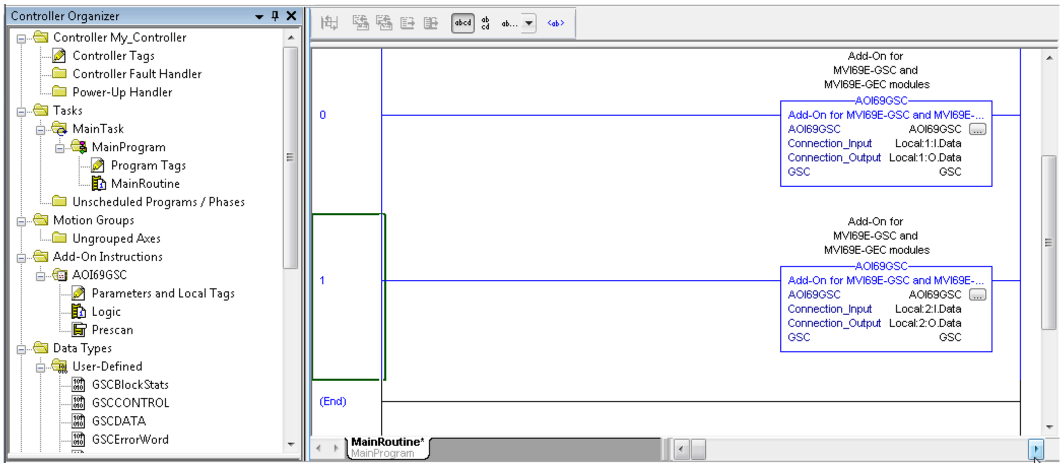
12 Associate the I/O connection variables to the correct module in the corresponding slot number. The default values are Local:1:I and Local:1:O. You must edit these values if the card is placed in a slot location other than slot 1 (Local:1:x means the card is located in slot 1). Since the second card is placed in slot 2, change the **FINAL NAME** to Local:2:I and Local:2:O. Also, you can append a '_2' at the end of the **FINAL NAME** of 'AOI69_GSC' and 'GSC' arrays as shown below.



Configure Tag References

Import Name	Operation	Final Name	Alias For	Data Type	Description
AOI69GSC	Use Existing	AOI69GSC		AOI69GSC	
GSC	Use Existing	GSC		GSCModuleDef	
Local:1:I	Use Existing	Local:2:I		PS:MVI69_M...	
Local:1:O	Use Existing	Local:2:O		PS:MVI69_M...	

13 Click **OK** to confirm.



0

Add-On for
MVI69E-GSC and
MVI69E-GEC modules

AOI69GSC
Add-On for MVI69E-GSC and MVI69E-...
AOI69GSC AOI69GSC
Connection_Input Local:1:I.Data
Connection_Output Local:1:O.Data
GSC GSC

1

Add-On for
MVI69E-GSC and
MVI69E-GEC modules

AOI69GSC
Add-On for MVI69E-GSC and MVI69E-...
AOI69GSC AOI69GSC
Connection_Input Local:2:I.Data
Connection_Output Local:2:O.Data
GSC GSC

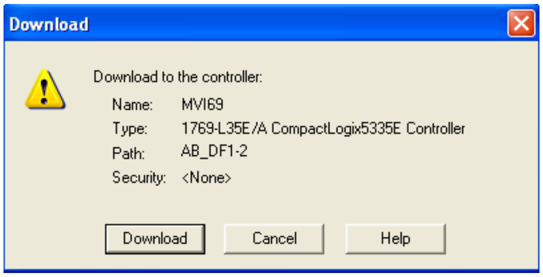
(End)

14 The setup procedure is now complete. Save and download the project to the processor.

2.3 Downloading the Sample Program to the Processor

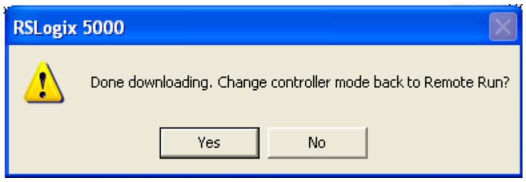
Note: The key switch on the front of the *CompactLogix* processor must be in the REM or PROG position.

1. If you are not already online to the processor, open the **COMMUNICATIONS** menu, and then choose **DOWNLOAD.** *RSLogix* will establish communication with the processor.
2. When communication is established, *RSLogix* will open a confirmation dialog box. Click the **DOWNLOAD** button to transfer the sample program to the processor.



Download to the controller:
Name: MVI69
Type: 1769-L35E/A CompactLogix5335E Controller
Path: AB_DF1-2
Security: <None>

3. *RSLogix* will compile the program and transfer it to the processor. This process may take a few minutes.
4. When the download is complete, *RSLogix* will open another confirmation dialog box. Click **OK** to switch the processor from *PROGRAM* mode to *RUN* mode.



Done downloading. Change controller mode back to Remote Run?

Note: If you receive an error message during these steps, refer to your *RSLogix* documentation to interpret and correct the error.

3 Optional AOI

The Optional AOI supports the following optional features:

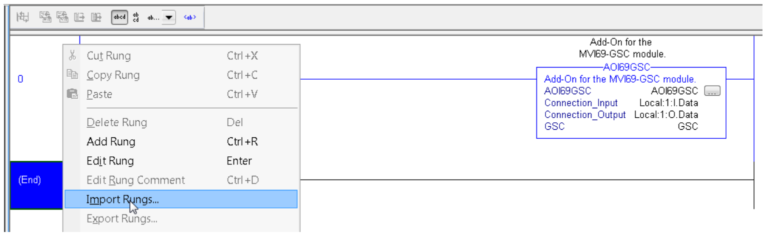
- Read/Write IP Address
- Read/Write Date Time

Using controller tags, the Optional AOI allows you to request and set the module's IP address, date, and time. These optional features are not supported by the MVI69-GSC legacy module.

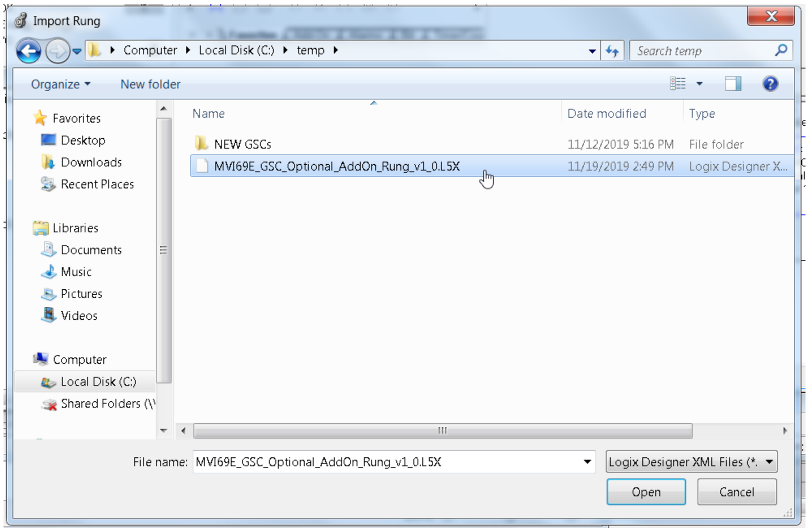
> **Note:** The Optional AOI may be added to an existing legacy MVI69-GSC application to add the new functionality during a module replacement.

3.1 Importing the Optional AOI

1 Add a new rung to the existing processor ladder logic. Right-click on the new rung and select *Import Rungs…*

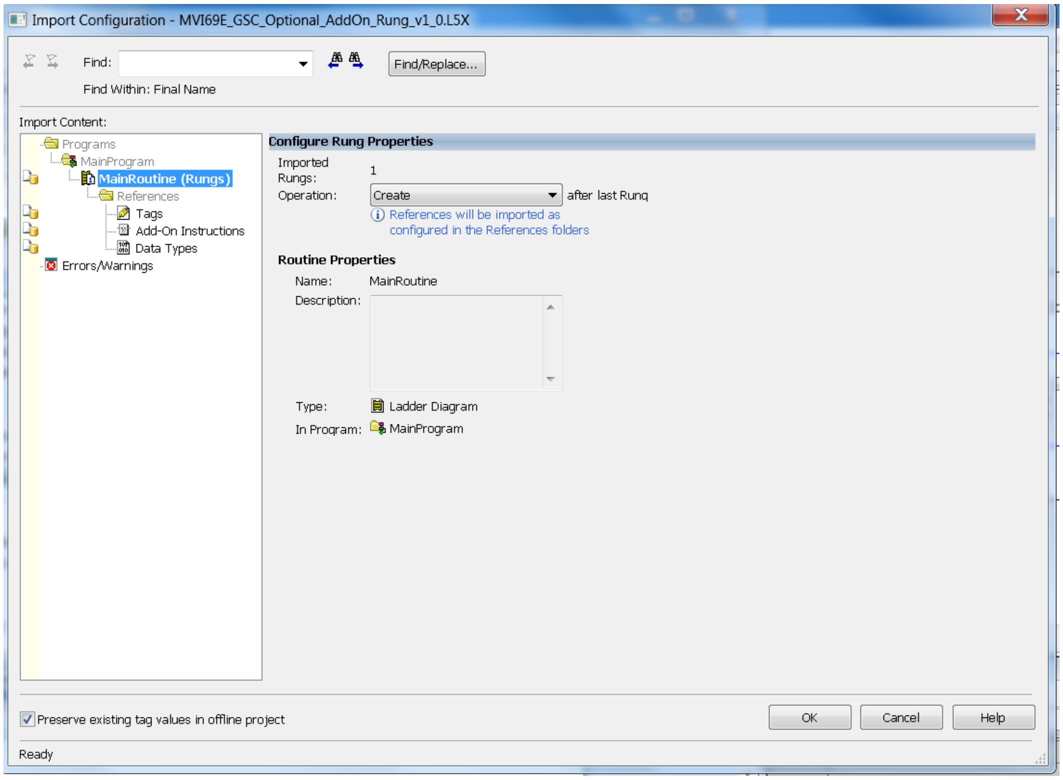


2 Select the Optional AOI file: *MVI69E_GSC_Optional_AddOn_Rung.L5X*



File name: MVI69E_GSC_Optional_AddOn_Rung_v1_0.L5X

3 At the *Import Configuration* window, select the *Operation* parameter to **CREATE**. Then click **OK**.


Import Configuration - MVI69E_GSC_Optional_AddOn_Rung_v1_0.L5X

Find:

Find Within: Final Name

Import Content:

Configure Rung Properties

Imported Rungs: 1

Operation: Create after last Rung

References will be imported as configured in the References folders

Routine Properties

Name: MainRoutine

Description:

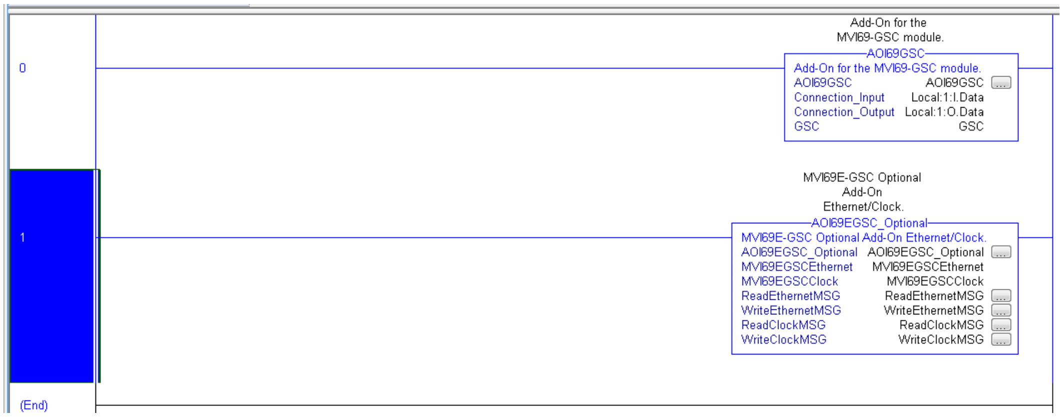
Type: Ladder Diagram

In Program: MainProgram

☑ Preserve existing tag values in offline project

Ready

4 The imported AOI rung is now in place.



Add-On for the MVI69-GSC module.

AOI69GSC

AOI69GSC	AOI69GSC
Connection_Input	Local:1:I.Data
Connection_Output	Local:1:O.Data
GSC	GSC

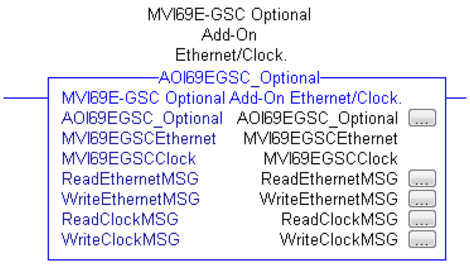
MVI69E-GSC Optional Add-On Ethernet/Clock.

AOI69EGSC_Optional

AOI69EGSC_Optional	AOI69EGSC_Optional
MVI69EGSCEthernet	MVI69EGSCEthernet
MVI69EGSCClock	MVI69EGSCClock
ReadEthernetMSG	ReadEthernetMSG
WriteEthernetMSG	WriteEthernetMSG
ReadClockMSG	ReadClockMSG
WriteClockMSG	WriteClockMSG

3.2 Setting up the Optional AOI

1 Click on the *ReadEthernetMSG* [...] icon to configure the message route:



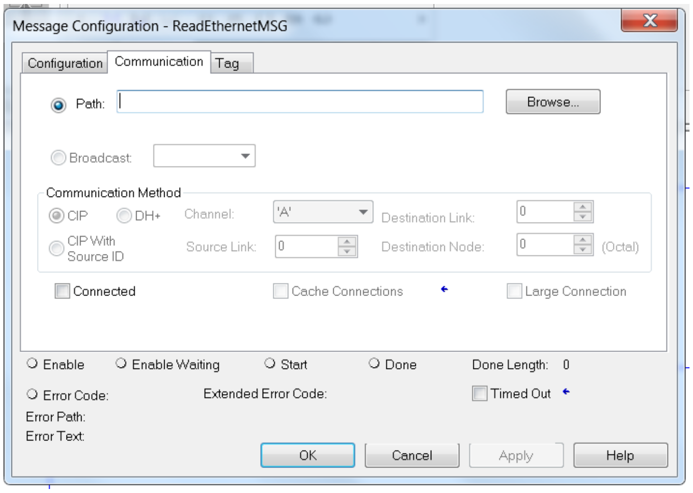
MVI69E-GSC Optional Add-On Ethernet/Clock.

AOI69EGSC_Optional

MVI69E-GSC Optional Add-On Ethernet/Clock.

AOI69EGSC_Optional	AOI69EGSC_Optional [...]
MVI69EGSCEthernet	MVI69EGSCEthernet
MVI69EGSCClock	MVI69EGSCClock
ReadEthernetMSG	ReadEthernetMSG [...]
WriteEthernetMSG	WriteEthernetMSG [...]
ReadClockMSG	ReadClockMSG [...]
WriteClockMSG	WriteClockMSG [...]

2 In the *Message Configuration* dialog, under the *Communication* tab, select the **BROWSE** button.



Message Configuration - ReadEthernetMSG

Configuration | Communication | Tag

● Path: [ ] Browse...

○ Broadcast:

Communication Method

○ CIP ○ DH+ Channel: 'A' Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected □ Cache Connections □ Large Connection

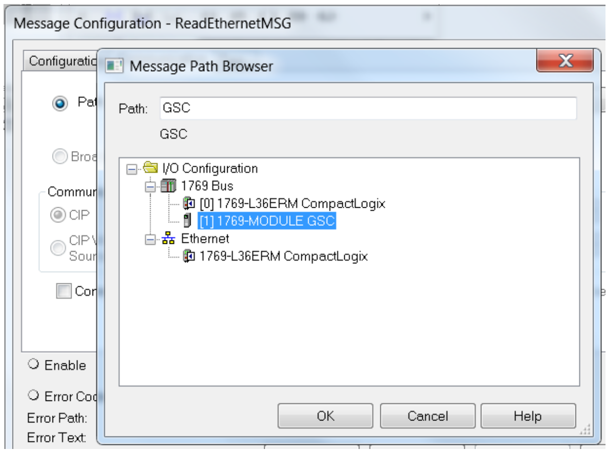
○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

3 Select the MVI69E-GSC module configured at the 1769 Bus and click at **OK**.



Message Configuration - ReadEthernetMSG

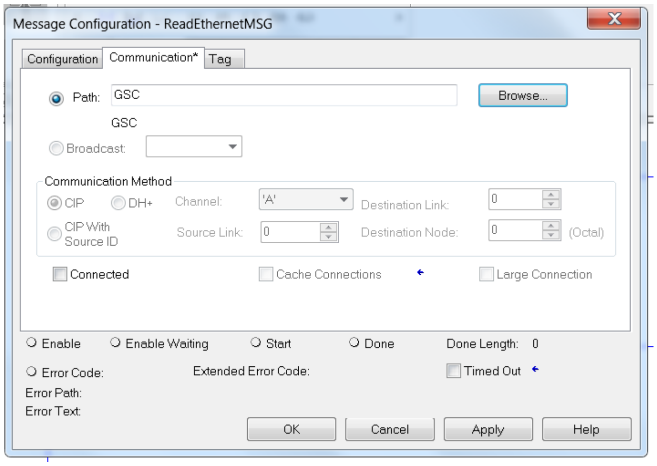
Message Path Browser

Path: GSC

GSC

```
⊟ I/O Configuration
  ⊟ 1769 Bus
      [0] 1769-L36ERM CompactLogix
      [1] 1769-MODULE GSC
  ⊟ Ethernet
      1769-L36ERM CompactLogix
```

4 The module name is displayed at the path field. Select **OK** to confirm the route configuration.



● Path: GSC

GSC

○ Broadcast:

Communication Method

● CIP ○ DH+ Channel: 'A' Destination Link: 0

○ CIP With Source ID Source Link: 0 Destination Node: 0 (Octal)

□ Connected □ Cache Connections □ Large Connection

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

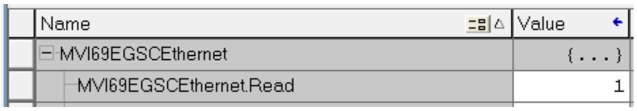
5 Repeat the same procedure to set the route for the remaining messages:

- *WriteEthernetMSG* ...
- *ReadClockMSG* ...
- *WriteClockMSG* ...

3.3 Synchronizing the IP Settings from the MVI69E-GSC to the Processor

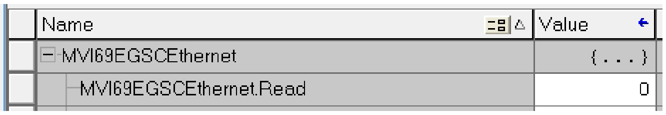
This section covers the process to read the IP settings from the MVI69E-GSC and implement them in the processor.

1 To trigger the IP settings read operation, set the *MVI69EGSCEthernet.Read* bit to '**1**'.



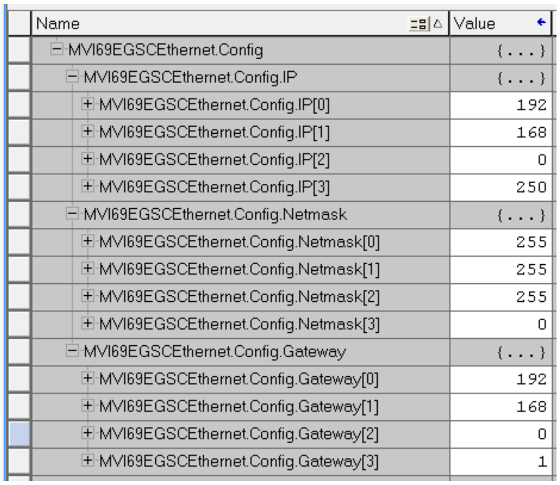
Name	Value
MVI69EGSCEthernet	{...}
MVI69EGSCEthernet.Read	1

2 Once the operation is concluded, the tag will be automatically reset to '**0**'



Name	Value
MVI69EGSCEthernet	{...}
MVI69EGSCEthernet.Read	0

3 The data is stored in the *MVI69EGSCEthernet.Config* tags (IP, Netmask, Gateway) as follows:

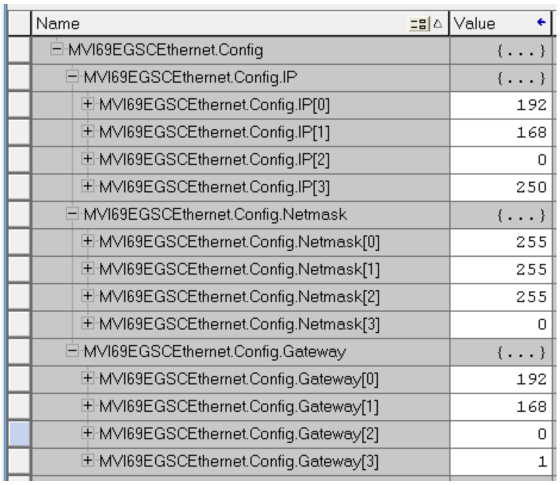


Name	Value
MVI69EGSCEthernet.Config	{...}
MVI69EGSCEthernet.Config.IP	{...}
MVI69EGSCEthernet.Config.IP[0]	192
MVI69EGSCEthernet.Config.IP[1]	168
MVI69EGSCEthernet.Config.IP[2]	0
MVI69EGSCEthernet.Config.IP[3]	250
MVI69EGSCEthernet.Config.Netmask	{...}
MVI69EGSCEthernet.Config.Netmask[0]	255
MVI69EGSCEthernet.Config.Netmask[1]	255
MVI69EGSCEthernet.Config.Netmask[2]	255
MVI69EGSCEthernet.Config.Netmask[3]	0
MVI69EGSCEthernet.Config.Gateway	{...}
MVI69EGSCEthernet.Config.Gateway[0]	192
MVI69EGSCEthernet.Config.Gateway[1]	168
MVI69EGSCEthernet.Config.Gateway[2]	0
MVI69EGSCEthernet.Config.Gateway[3]	1

3.4 Synchronizing the IP Settings from the Processor to the MVI69E-GSC

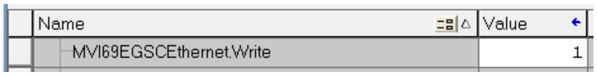
This section covers the process of sending the IP settings from the processor to the MVI69E-GSC.

1 Populate the IP settings in the *MVI69EGSCEthernet.Config* tag:



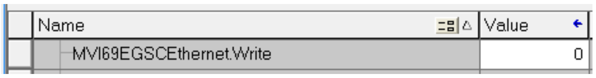
Name	Value
MVI69EGSCEthernet.Config	{...}
MVI69EGSCEthernet.Config.IP	{...}
MVI69EGSCEthernet.Config.IP[0]	192
MVI69EGSCEthernet.Config.IP[1]	168
MVI69EGSCEthernet.Config.IP[2]	0
MVI69EGSCEthernet.Config.IP[3]	250
MVI69EGSCEthernet.Config.Netmask	{...}
MVI69EGSCEthernet.Config.Netmask[0]	255
MVI69EGSCEthernet.Config.Netmask[1]	255
MVI69EGSCEthernet.Config.Netmask[2]	255
MVI69EGSCEthernet.Config.Netmask[3]	0
MVI69EGSCEthernet.Config.Gateway	{...}
MVI69EGSCEthernet.Config.Gateway[0]	192
MVI69EGSCEthernet.Config.Gateway[1]	168
MVI69EGSCEthernet.Config.Gateway[2]	0
MVI69EGSCEthernet.Config.Gateway[3]	1

2 Set the *MVI69EGSCEthernet.Write* bit to '**1**' to trigger the IP settings write operation.



Name	Value
MVI69EGSCEthernet.Write	1

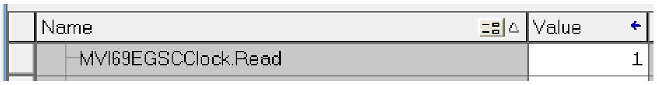
3 The *MVI69EGSCEthernet.Write* bit will be automatically reset once the operation is concluded.



Name	Value
MVI69EGSCEthernet.Write	0

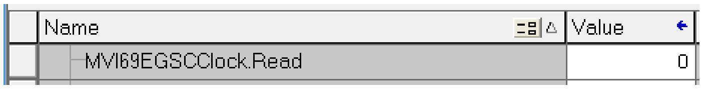
3.5 Reading the Date/Time from the MVI69E-GSC to the Processor

1 Toggle the *MVI69EGSCClock.Read* bit to '**1**' to toggle the date/time read operation.



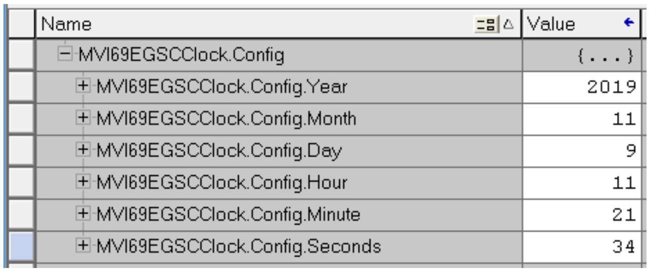
Name	Value
MVI69EGSCClock.Read	1

2 The *MVI69EGSCClock.Read* bit will be automatically reset once the operation is concluded.



Name	Value
MVI69EGSCClock.Read	0

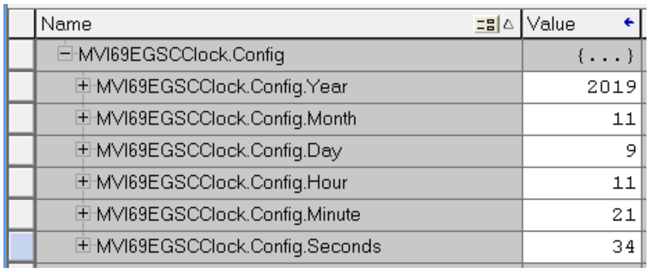
3 The date and time read from the MVI69E-GSC is stored at the *MVI69EGSCClock.Config* tag.



Name	Value
MVI69EGSCClock.Config	{...}
MVI69EGSCClock.Config.Year	2019
MVI69EGSCClock.Config.Month	11
MVI69EGSCClock.Config.Day	9
MVI69EGSCClock.Config.Hour	11
MVI69EGSCClock.Config.Minute	21
MVI69EGSCClock.Config.Seconds	34

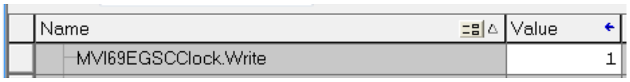
3.6 Writing the Date/Time from the Processor to the MVI69E-GSC

1 Populate date and time values in the *MVI69EGSCClock.Config* tag.



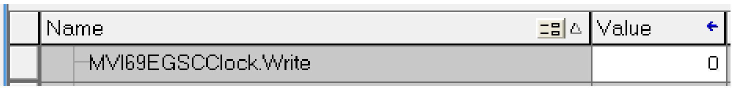
Name	Value
MVI69EGSCClock.Config	{...}
MVI69EGSCClock.Config.Year	2019
MVI69EGSCClock.Config.Month	11
MVI69EGSCClock.Config.Day	9
MVI69EGSCClock.Config.Hour	11
MVI69EGSCClock.Config.Minute	21
MVI69EGSCClock.Config.Seconds	34

2 Toggle the *MVI69EGSCClock.Write* bit to '**1**' to trigger the write date/time operation.



Name	Value
MVI69EGSCClock.Write	1

3 The *MVI69EGSCClock.Write* tag will be automatically reset once the write date/time operation is concluded.



Name	Value
MVI69EGSCClock.Write	0

4 Using ProSoft Configuration Builder

ProSoft Configuration Builder (PCB) provides a quick and easy way to manage module configuration files customized to meet your application needs. *PCB* is not only a powerful solution for new configuration files, but also allows you to import information from previously installed (known working) configurations to new projects.

4.1 Installing ProSoft Configuration Builder Software

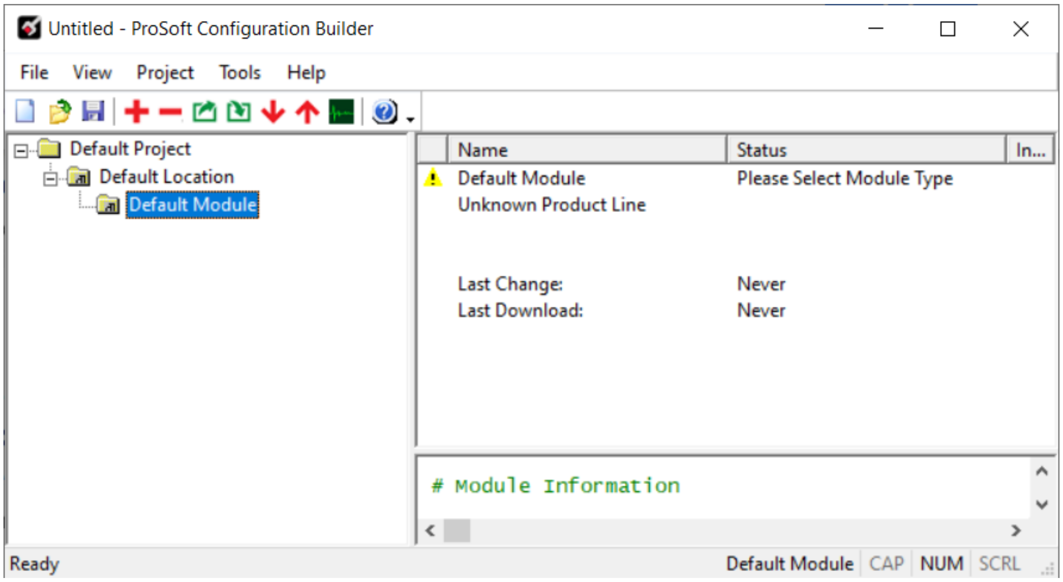
You must install the *ProSoft Configuration Builder (PCB)* software to configure the module. You can download the newest version of *ProSoft Configuration Builder* from www.prosoft-technology.com.

Installing ProSoft Configuration Builder from the ProSoft website

1 Open your web browser and navigate to *http://www.prosoft-technology.com/pcb*
2 Download the latest version of *ProSoft Configuration Builder*.
3 Choose **SAVE** or **SAVE FILE** when prompted.
4 Save the file to your *Windows Desktop*, so that you can find it easily when you have finished downloading.
5 When the download is complete, locate and open the file, and then follow the instructions on your screen to install the program.

4.2 Setting Up the Project

ProSoft Configuration Builder's window consists of a tree view on the left, an information pane and a configuration pane on the right side of the window.

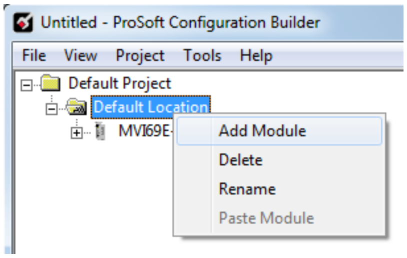

Untitled - ProSoft Configuration Builder

Name	Status	In...
Default Module	Please Select Module Type	
Unknown Product Line		
Last Change:	Never	
Last Download:	Never	

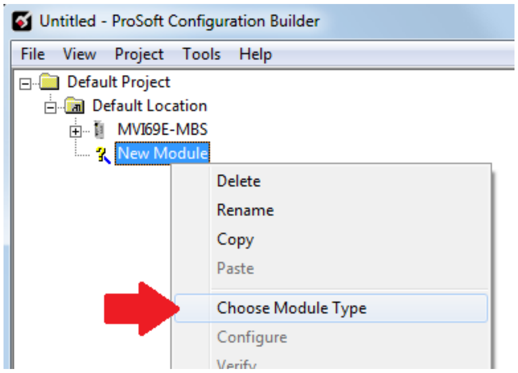
```
# Module Information
```

4.2.1 Adding the MVI69E-GSC module to the project.

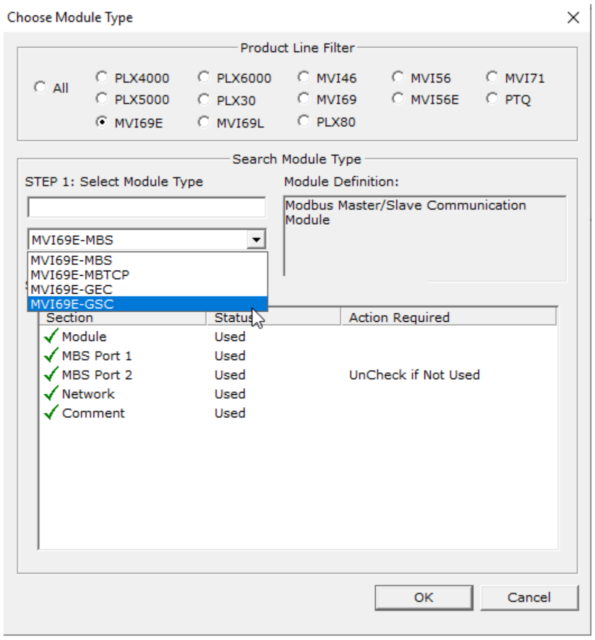
1 Right click **DEFAULT LOCATION** (which you can rename) and choose **ADD MODULE**.



2 Right-click **NEW MODULE** and choose **CHOOSE MODULE TYPE**.



3 In the *Choose Module Type* dialog box, select **MVI69E** in the **PRODUCT LINE FILTER** area, and then select **MVI69E-GSC** as the **MODULE TYPE**. Click **OK**.

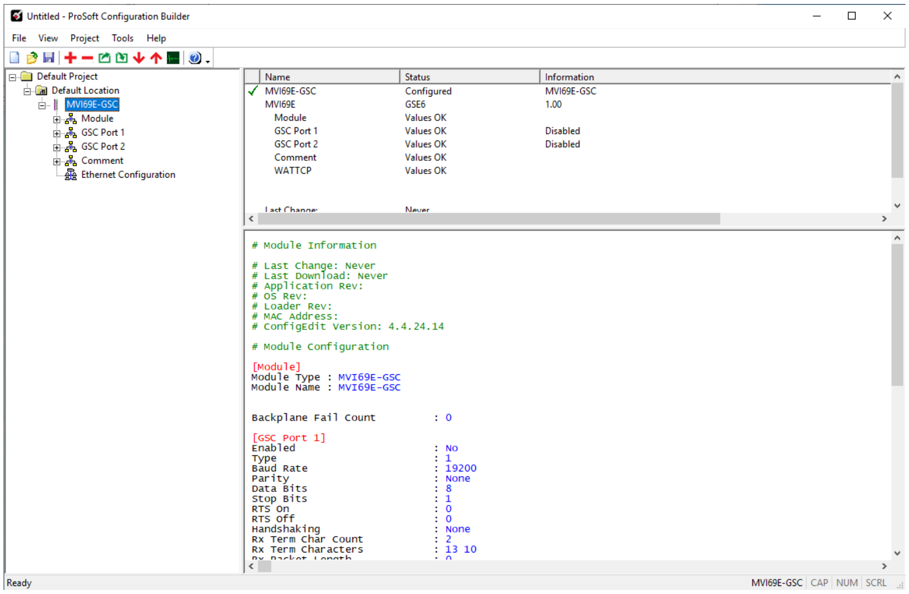

Choose Module Type
Product Line Filter
○ All ○ PLX4000 ○ PLX6000 ○ MVI46 ○ MVI56 ○ MVI71
○ PLX5000 ○ PLX30 ○ MVI69 ○ MVI56E ○ PTQ
● MVI69E ○ MVI69L ○ PLX80
Search Module Type
STEP 1: Select Module Type
Module Definition:
MVI69E-MBS
MVI69E-MBS
MVI69E-MBTCP
MVI69E-GEC
MVI69E-GSC
OK
Cancel

Modbus Master/Slave Communication Module

Section	Status	Action Required
✓ Module	Used	
✓ MBS Port 1	Used	
✓ MBS Port 2	Used	UnCheck if Not Used
✓ Network	Used	
✓ Comment	Used	

4.3 Renaming PCB Objects

Notice that the contents of the information pane and the configuration pane changed when you added the module to the project.



- Default Project
  - Default Location
    - MVI69E-GSC
      - Module
      - GSC Port 1
      - GSC Port 2
      - Comment
      - Ethernet Configuration

	Name	Status	Information
✓	MVI69E-GSC	Configured	MVI69E-GSC
	MVI69E	GSE6	1.00
	Module	Values OK	
	GSC Port 1	Values OK	Disabled
	GSC Port 2	Values OK	Disabled
	Comment	Values OK	
	WATTCP	Values OK	
	Last Change:	Never	

```
# Module Information

# Last Change: Never
# Last Download: Never
# Application Rev:
# OS Rev:
# Loader Rev:
# MAC Address:
# ConfigEdit Version: 4.4.24.14

# Module Configuration

[Module]
Module Type : MVI69E-GSC
Module Name : MVI69E-GSC


Backplane Fail Count        : 0

[GSC Port 1]
Enabled                     : No
Type                        : 1
Baud Rate                   : 19200
Parity                      : None
Data Bits                   : 8
Stop Bits                   : 1
RTS On                      : 0
RTS Off                     : 0
Handshaking                 : None
Rx Term Char Count          : 2
Rx Term Characters          : 13 10
Rx Packet Length            : 0
```

At this time, you may wish to rename the *Default Project* and *Default Location* folders in the tree view.

1 Select the object, and then right-click the mouse button to open a shortcut menu. From the shortcut menu, choose **RENAME.**
2 Type the name to assign to the object.
3 Click *away* from the object to save the new name.

<u>Configuring Module Parameters</u>

1 Click on the **[+]** sign next to the module icon to expand module information.
2 Click on the **[+]** sign next to any icon to view module information and configuration options.
3 Double-click any icon to open an *Edit* dialog box.
4 To edit a parameter, select the parameter in the left pane and make your changes in the right pane.
5 Click **OK** to save your changes.

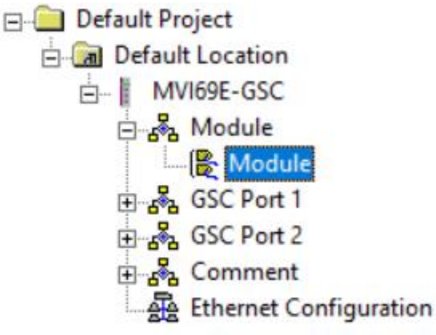
<u>Printing a Configuration File</u>

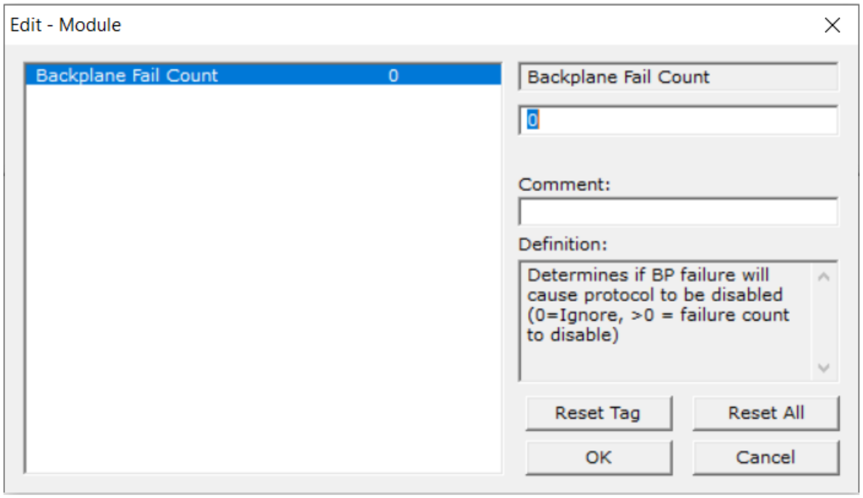
1 Select the module icon, and right-click the mouse button to open a shortcut menu.
2 On the shortcut menu, choose **VIEW CONFIGURATION.** This action opens the *View Configuration* window.
3 In the *View Configuration* window, open the **FILE** menu, and choose **PRINT.** This action opens the *Print* dialog box.
4 In the *Print* dialog box, choose the printer to use from the drop-down list, select printing options, and then click **OK.**

4.4 Module Configuration Parameters

This section describes the MVI69E-GSC parameters configured in ProSoft Configuration Builder.

4.4.1 Module Parameters




Edit - Module

Backplane Fail Count 0

Backplane Fail Count

0

Comment:

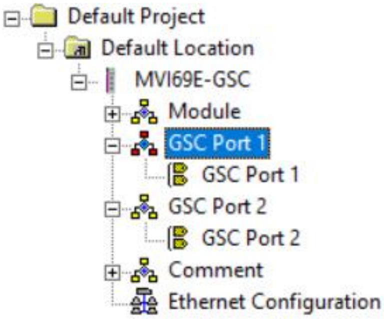
Definition:

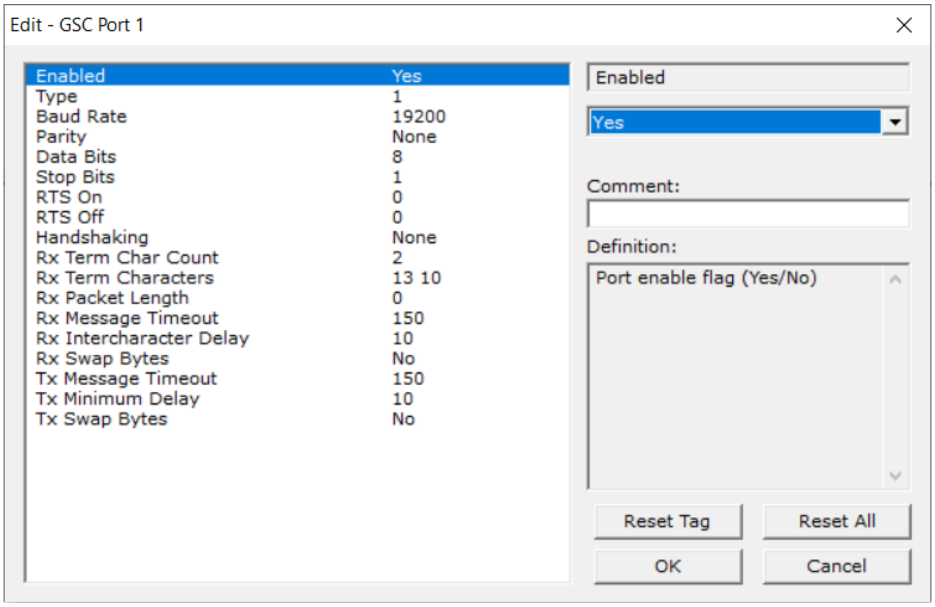
Determines if BP failure will cause protocol to be disabled (0=Ignore, >0 = failure count to disable)

[Section]/Item	Range	Description
[MODULE]		Module section header
Backplane Fail Count:	0 to 65535	This parameter specifies the number of successive transfer errors that must occur before the communication ports are shut down. If the parameter is set to zero, the communication ports will continue to operate under all conditions. If the value is set larger than 0 (1 to 65535), communications will cease if the specified number of backplane failures occur.

4.4.2 GSC Port x Parameters

The *GSC Port 1* and *2*, respectively, have identical port parameters.





Edit - GSC Port 1

Enabled	Yes
Type	1
Baud Rate	19200
Parity	None
Data Bits	8
Stop Bits	1
RTS On	0
RTS Off	0
Handshaking	None
Rx Term Char Count	2
Rx Term Characters	13 10
Rx Packet Length	0
Rx Message Timeout	150
Rx Intercharacter Delay	10
Rx Swap Bytes	No
Tx Message Timeout	150
Tx Minimum Delay	10
Tx Swap Bytes	No

Enabled

Yes

Comment:

Definition:

Port enable flag (Yes/No)

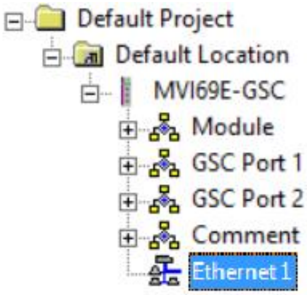
Parameter	Range	Description
[GSC Port 1 or 2]	[redacted]	MVI69E-GSC port definition header
Enabled	Yes or No	This parameter defines if this port will be utilized.
Type	0 to 15	This parameter specifies the receive termination characteristics for the port. This value is bit mapped as follows: Bit 0 = Termination character(s) used Bit 1 = Message timeout used Bit 2 = Intercharacter delay timeout used Bit 3 = Packet size limit used. If the parameter is set to zero, the port is placed in stream mode.
Baud Rate	From selected list of codes	This is the baud rate to be used on the port. Enter the baud rate as a value. For example, to select 19K baud, enter 19200. Valid entries for this field include: 110, 150, 300, 600, 1200, 2400, 4800, 9600, 19200, 384 or 3840 for 38400, 576 or 5760 for 57600 and 115 or 1150 for 115200.

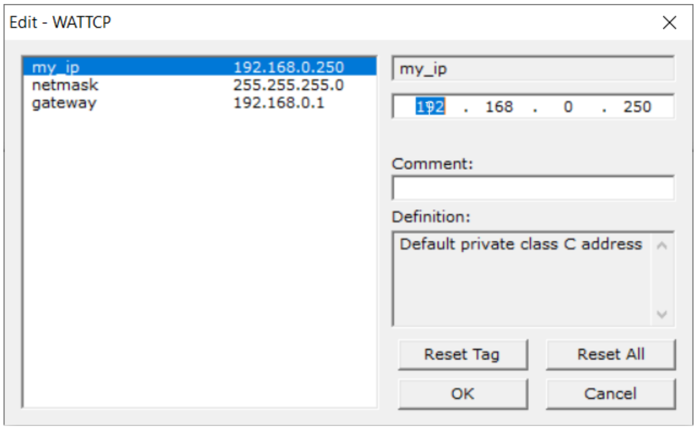
Parameter	**Range**	**Description**
Parity	None, Even, Odd, Mark or Space	This is the Parity code to be used for the port.
Data Bits	5 to 8	This parameter sets the number of data bits for each word used by the protocol.
Stop Bits	1 or 2	This parameter sets the number of stop bits to be used with each data value sent.
RTS On	0 to 65535	This parameter sets the number of milliseconds to delay after RTS is asserted before the data will be transmitted.
RTS Off	0 to 65535	This parameter sets the number of milliseconds to delay after the last byte of data is sent before the RTS modem signal will be set low.
Handshaking	NONE, RTS/CTS, DTR/DSR or XON/XOFF	This parameter specifies the handshaking used on the port. The values are as follows: None = No hardware or software handshaking RTS/CTS = Hardware handshaking DTR/DSR = Hardware handshaking XON/XOFF = Software handshaking
Rx Term Char Count	0 to 12	This parameter is used if bit 0 of the *Type* parameter is set. This value (0 to 12) defines the number of termination characters used to define the end of received message.
Rx Term Characters	List of up to 12 integer values	This array of 12 integer values representing the characters used to define the termination characters at the end of each received message. The number of characters to be used in the array is set in the *RTermCnt* parameter.
Rx Packet Length	0 to 4096	This parameter is used if bit 3 is set in the *Type* parameter. The parameter sets the length of data required to be received on the port before transferring the data to the processor.
Rx Message Timeout	0 to 65535	This parameter is used if bit 1 is set in the *Type* parameter. The parameter sets the number of milliseconds to wait after the first character is received on the port before automatically sending the data to the processor.
Rx Intercharacter Delay	0 to 65535	This parameter is used if bit 2 is set in the *Type* parameter. The parameter sets the number of milliseconds to wait between each character received on the port before sending the data to the processor.
Rx Swap Bytes	Yes or No	This parameter specifies if the data received should have its bytes swapped before sending over the backplane.
Tx Message Timeout	0 to 65535	This parameter specifies the timeout period to transmit a message out the port. A message must be transmitted out the port within the specified timeout period. Message transmission will be aborted if the timeout is exceeded.
Tx Minimum Delay	0 to 65535	This parameter specifies the minimum number of milliseconds to delay before transmitting a message out the port. This pre-send delay is applied before the RTS on time. This may be required when communicating with slow devices.
Tx Swap Bytes	Yes or No	This parameter specifies if the data to be transmitted out the port will have the bytes swapped from the data presented across the backplane.

4.4.3 MVI69E-GSC IP Address Configuration

This section defines the permanent IP address, Subnet Mask, and Gateway of the module.

1 In the ProSoft Configuration Builder tree view, double-click the **ETHERNET 1** icon.





my_ip 192.168.0.250
netmask 255.255.255.0
gateway 192.168.0.1

my_ip

192 . 168 . 0 . 250

Comment:

Definition:

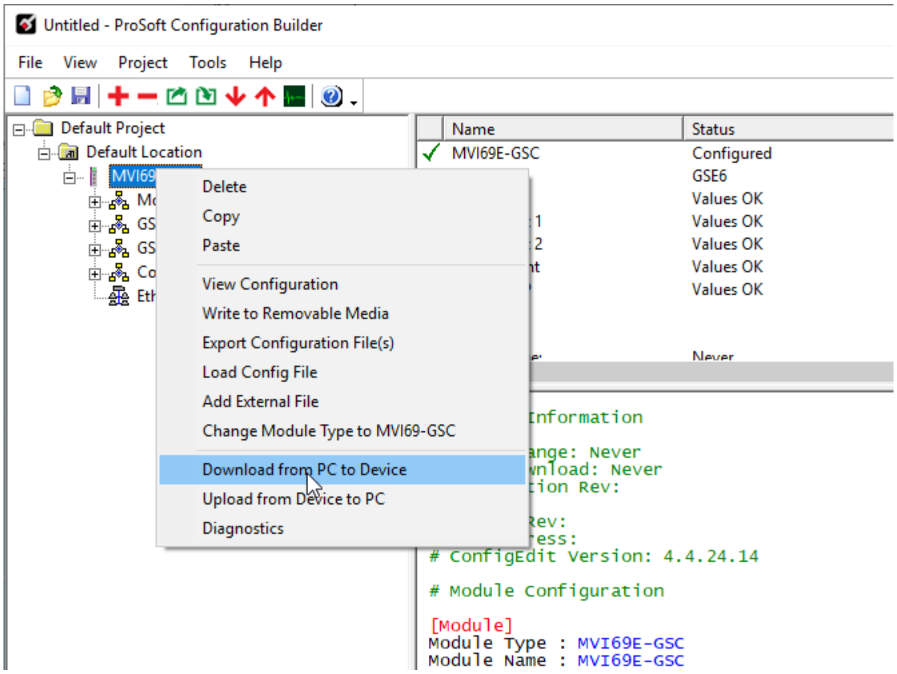
Default private class C address

Parameter	Description
My_ip	Unique IP address assigned to the module
Netmask	Subnet mask of module
Gateway	Gateway (if used)

2 Click **OK** when complete.

4.5 Downloading the Configuration File to the Module

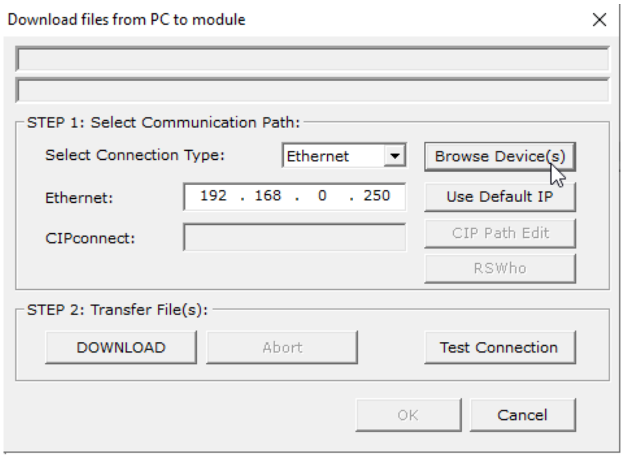
1 In the ProSoft Configuration Builder tree view, right-click the module icon and choose **DOWNLOAD FROM PC TO DEVICE**.



Name	Status
MVI69E-GSC	Configured
	GSE6
	Values OK
1	Values OK
2	Values OK
t	Values OK
	Values OK
e:	Never

```
Information

ange: Never
wnload: Never
tion Rev:

Rev:
ress:
# ConfigEdit Version: 4.4.24.14

# Module Configuration

[Module]
Module Type : MVI69E-GSC
Module Name : MVI69E-GSC
```

2 In the *Download Files from PC to Module* dialog box, select **BROWSE DEVICES**.

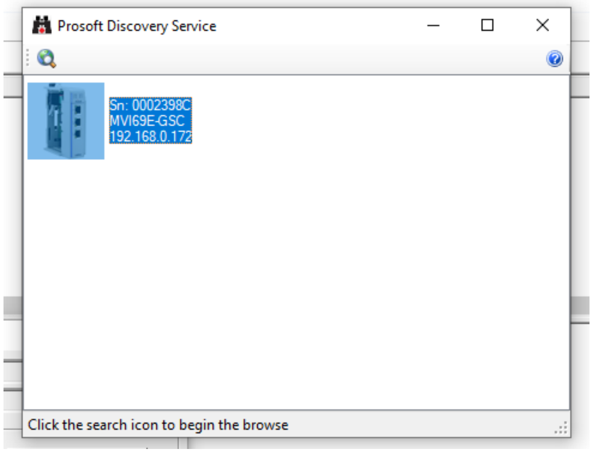

Download files from PC to module
STEP 1: Select Communication Path:
Browse Device(s)
Use Default IP
CIP Path Edit
RSWho
STEP 2: Transfer File(s):
DOWNLOAD
Abort
Test Connection
OK
Cancel

Select Connection Type: Ethernet

Ethernet: 192 . 168 . 0 . 250

CIPconnect:

3 In the *ProSoft Discovery Service* dialog box, select the **MVI69E-GSC** module:

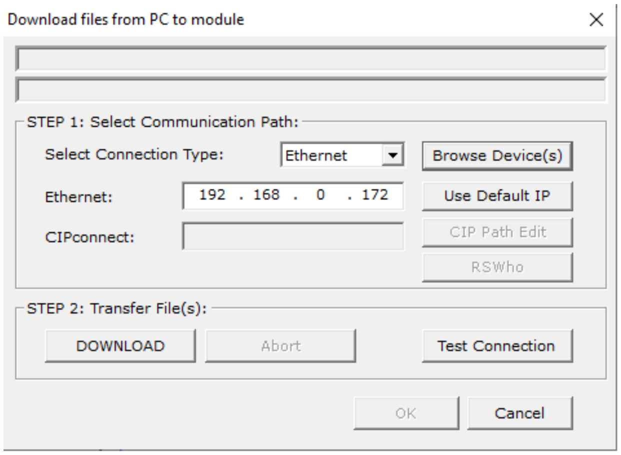


Prosoft Discovery Service

Sn: 0002398C
MVI69E-GSC
192.168.0.172

Click the search icon to begin the browse

4 The *IP address* field is populated with the module's IP address. Click **DOWNLOAD** to confirm the operation:



Download files from PC to module

STEP 1: Select Communication Path:

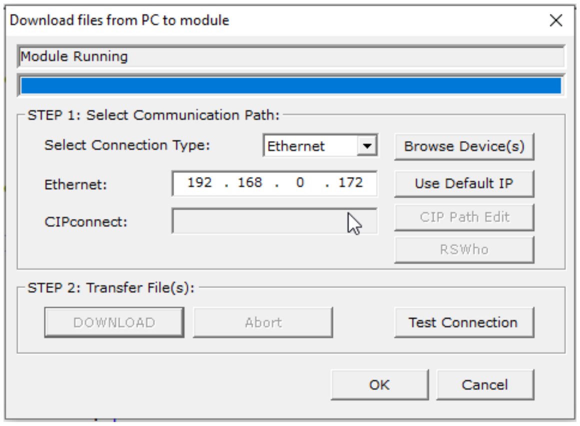
Select Connection Type: Ethernet

Ethernet: 192 . 168 . 0 . 172

CIPconnect:

STEP 2: Transfer File(s):

5 Once concluded, the status field shows 'Module Running'. Click **OK**.



Download files from PC to module

Module Running

STEP 1: Select Communication Path:

Select Connection Type: Ethernet

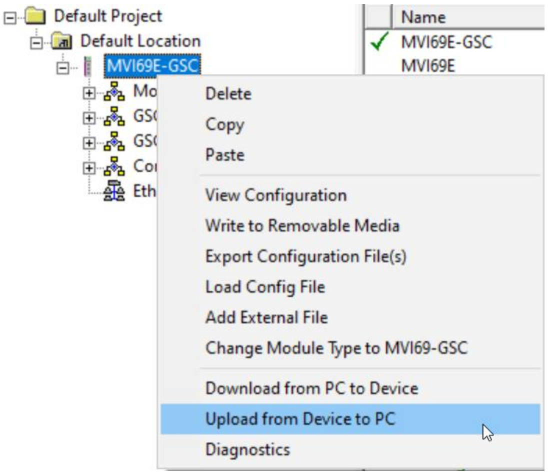
Ethernet: 192 . 168 . 0 . 172

CIPconnect:

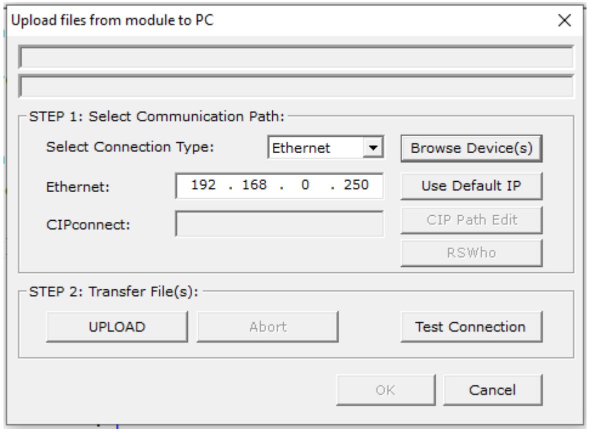
STEP 2: Transfer File(s):

4.6 Uploading the Configuration File from the Module

1 In the ProSoft Configuration Builder tree view, right-click the **MVI69E-GSC** icon and choose **UPLOAD FROM DEVICE TO PC**.



2 Click the **BROWSE DEVICE(S)** button.

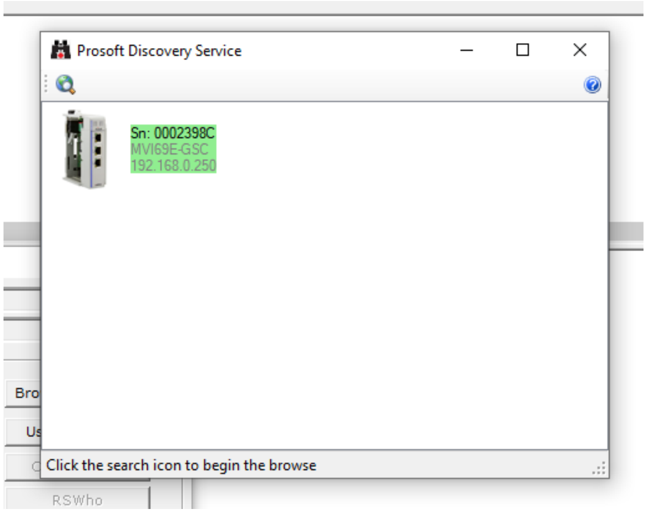

Upload files from module to PC
STEP 1: Select Communication Path:
Browse Device(s)
Use Default IP
CIP Path Edit
RSWho
STEP 2: Transfer File(s):
UPLOAD
Abort
Test Connection
OK
Cancel

Select Connection Type: Ethernet

Ethernet: 192 . 168 . 0 . 250

CIPconnect:

3 In the *ProSoft Discovery Service* dialog box, select the MVI69E-GSC module. The *Ethernet* field will be automatically populated with the module's IP address.

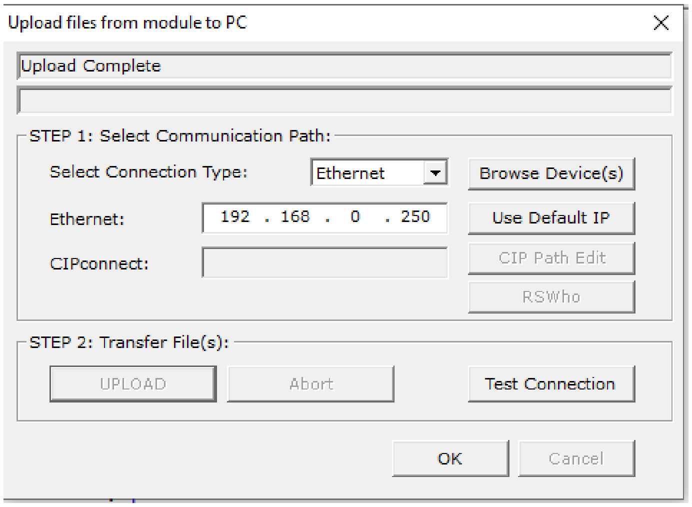


Prosoft Discovery Service

Sn: 0002398C
MVI69E-GSC
192.168.0.250

Click the search icon to begin the browse

4 Once the file is uploaded, the following message is displayed:



Upload files from module to PC

Upload Complete

STEP 1: Select Communication Path:

Select Connection Type: Ethernet

Ethernet: 192 . 168 . 0 . 250

CIPconnect:

STEP 2: Transfer File(s):

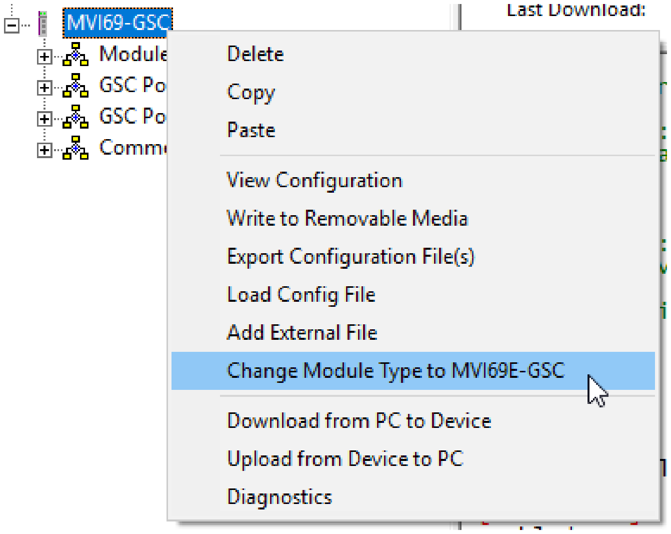
5 PCB now displays the uploaded configuration file.

4.7 Converting a Legacy MVI69-GSC to an MVI69E-GSC

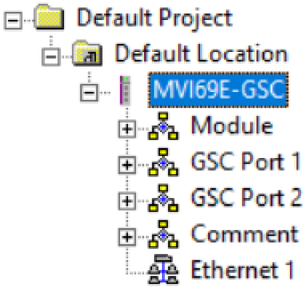
The MVI69E-GSC product is backward compatible with the legacy MVI69-GSC product. You may replace the existing MVI69-GSC module with a new MVI69E-GSC without requiring any modifications to the existing ladder logic.

This section shows how to convert an existing MVI69-GSC project to a MVI69E-GSC.

1 In PCB, right-click on the module and select **CHANGE MODULE TYPE TO MVI69E-GSC**.



2 The legacy project will be converted to the MVI69E-GSC project with the same module settings:



3 The new configuration file can now be downloaded to the MVI69E-GSC.

5 Using Controller Tags

The MVI69E-GSC data is stored in controller tags in the ladder logic. Controller tags are used to manage the communications between the MVI69E-GSC module and the CompactLogix or MicroLogix 1500-LRP processor:

- View the read and write data being transferred between the module and the processor.
- View status data for the module.
- Set up and trigger special functions.
- Initiate module restarts (Warm Boot or Cold Boot).

Individual controller tags can be grouped into collections of controller tags called controller tag structures. A controller tag structure can contain any combination of:

- Individual controller tags
- Controller tag arrays
- Lower-level controller tag structures

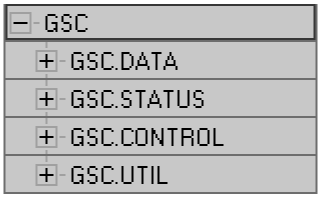
The controller tags are included in the MVI69E-GSC Add-On Instruction. After you import the Add-On Instruction, you can find the controller tags in the *Controller Tags* subfolder, located in the *Controller* folder in the *Controller Organizer* pane of the main Studio 5000 window.

The Add-On Instruction also includes user-defined data types (UDTs). UDTs are collections of data types and declares the data types for the controller tag structures.

The MVI69E-GSC Add-On Instruction is extensively commented to provide information on the purpose and function of each user-defined data type and controller tag. For most applications, the Add-On Instruction works without needing any modification.

5.1 MVI69E-GSC Controller Tags

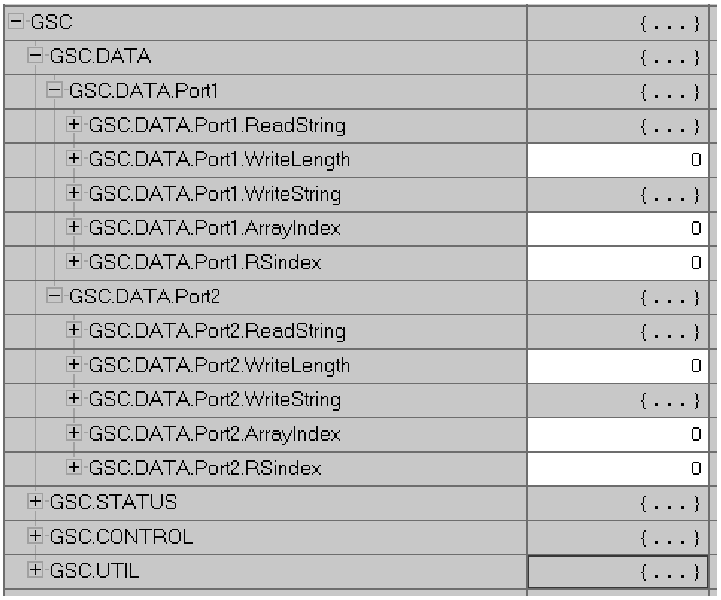
The main controller tag structure, *GSC*, is broken down into four lower-level controller tag structures.



- GSC
  - GSC.DATA
  - GSC.STATUS
  - GSC.CONTROL
  - GSC.UTIL

The four lower-level controller tag structures contain other controller tags and controller tag structures. Click the **[+]** sign next to any controller tag structure to expand it and view the next level in the structure.

For example, if you expand the *GSC.DATA* controller tag structure, it contains two controller tags; *GSC.DATA.Port1* and *GSC.DATA.Port2*, which stores the data received/sent to each port.



GSC	{...}
GSC.DATA	{...}
GSC.DATA.Port1	{...}
GSC.DATA.Port1.ReadString	{...}
GSC.DATA.Port1.WriteLength	0
GSC.DATA.Port1.WriteString	{...}
GSC.DATA.Port1.ArrayIndex	0
GSC.DATA.Port1.RSindex	0
GSC.DATA.Port2	{...}
GSC.DATA.Port2.ReadString	{...}
GSC.DATA.Port2.WriteLength	0
GSC.DATA.Port2.WriteString	{...}
GSC.DATA.Port2.ArrayIndex	0
GSC.DATA.Port2.RSindex	0
GSC.STATUS	{...}
GSC.CONTROL	{...}
GSC.UTIL	{...}

The controller tags in the Add-On Instruction are commented in the **DESCRIPTION** column.

5.2 User-Defined Data Types (UDTs)

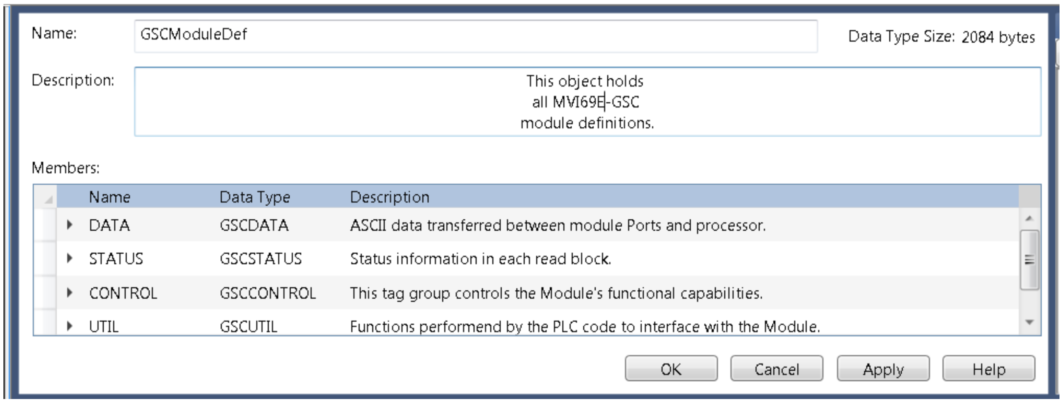
User-defined data types (UDTs) allow you to organize collections of data types into groupings. You can use these groupings, or data type structures, to declare the data types for controller tag structures. Another advantage of defining a UDT is that you may reuse it in other controller tag structures that use the same data types.

The Add-On Instruction for the MVI69E-GSC module has pre-defined UDTs. You can find them in the *User-Defined* subfolder, located in the *Data Types* folder in the *Controller Organizer* pane of the main RSLogix window. Like the controller tags, the UDTs are organized in a multiple-level tree structure.

5.2.1 MVI69E-GSC User-Defined Data Types

Several UDTs are defined for the MVI69E-GSC Add-On Instruction. The main UDT (*GSCModuleDef*) contains all the data types for the module. There are several UDTs that are one level below *GSCModuleDef*.

These lower-level UDTs were used to create the *GSC.DATA, GSC.STATUS, GSC***.** *CONTROL*, and *GSC.UTIL* controller tag structures.



Name: GSCModuleDef

Data Type Size: 2084 bytes

Description: This object holds all MVI69E-GSC module definitions.

Members:

Name	Data Type	Description
DATA	GSCDATA	ASCII data transferred between module Ports and processor.
STATUS	GSCSTATUS	Status information in each read block.
CONTROL	GSCCONTROL	This tag group controls the Module's functional capabilities.
UTIL	GSCUTIL	Functions performend by the PLC code to interface with the Module.

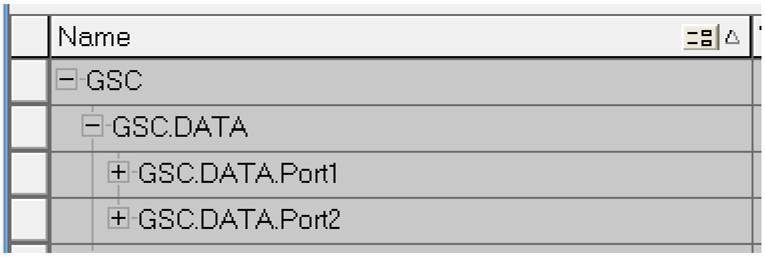
5.3 GSC Controller Tag Overview

This section details the *GSC* controller tags.

Tag Name	Description
GSC.DATA	Port 1 and Port 2 tags for data transfer
GSC.STATUS	Status information
GSC.CONTROL	Tasks that the processor may request from the module.
GSC.UTIL	Generic tags used for internal ladder processing (**DO NOT MODIFY**)

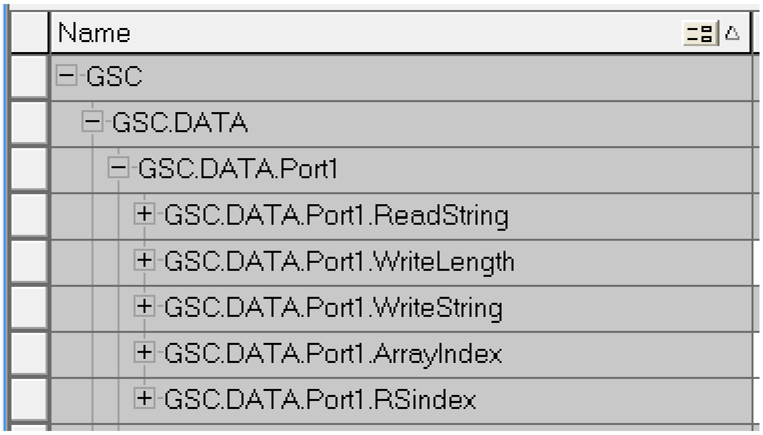
5.3.1 GSC.DATA

GSC.DATA contains the controller tags required for data exchange between the processor and the remote serial devices. This data contains one sub tag for each serial port (Port 1 and Port 2):



Name
- GSC
  - GSC.DATA
    - GSC.DATA.Port1
    - GSC.DATA.Port2

Each port tag contains the following controller tags for data exchange:

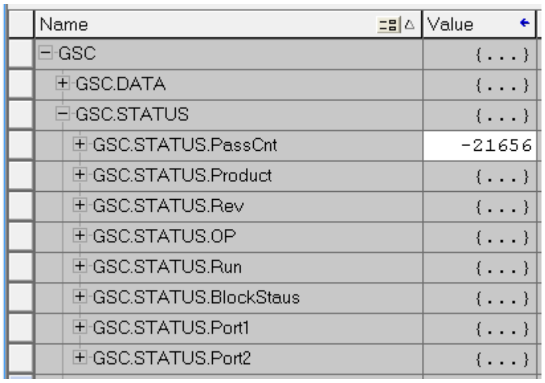


Name
- GSC
  - GSC.DATA
    - GSC.DATA.Port1
      - GSC.DATA.Port1.ReadString
      - GSC.DATA.Port1.WriteLength
      - GSC.DATA.Port1.WriteString
      - GSC.DATA.Port1.ArrayIndex
      - GSC.DATA.Port1.RSindex

Note: The *ArrayIndex* and *RSindex* tags are updated by the AOI logic to process the data transfer. Your application should not have to access these tags.

5.3.2 GSC.STATUS

The *GSC.STATUS* tag stores the module, backplane, and port diagnostics:



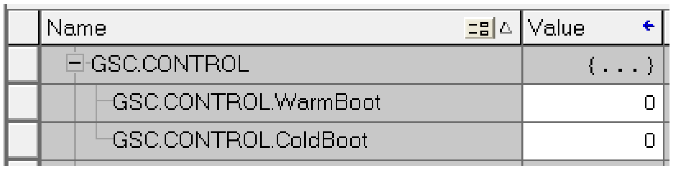
Name	Value
⊟ GSC	{...}
⊞ GSC.DATA	{...}
⊟ GSC.STATUS	{...}
⊞ GSC.STATUS.PassCnt	-21656
⊞ GSC.STATUS.Product	{...}
⊞ GSC.STATUS.Rev	{...}
⊞ GSC.STATUS.OP	{...}
⊞ GSC.STATUS.Run	{...}
⊞ GSC.STATUS.BlockStaus	{...}
⊞ GSC.STATUS.Port1	{...}
⊞ GSC.STATUS.Port2	{...}

For further information about each diagnostics tag, please refer to the *Status Data Definition* section.

5.3.3 GSC.CONTROL

The *GSC.CONTROL* tag contains all control tasks that may be requested by the processor to the module. The MVI69E-GSC supports two control operations: Warmboot and Coldboot.

To perform a module reboot (Warmboot or Coldboot), toggle the associated controller tag to ‘**1**’. The AOI logic will automatically reset the tag back to ‘**0**’.

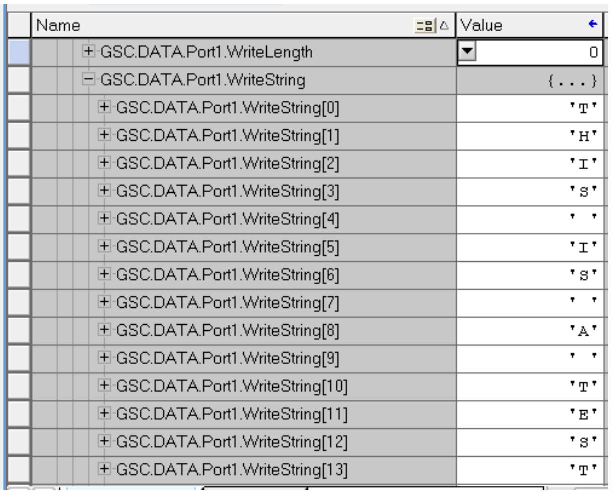


Name	Value
⊟ GSC.CONTROL	{...}
GSC.CONTROL.WarmBoot	0
GSC.CONTROL.ColdBoot	0

6 Transmitting ASCII Data

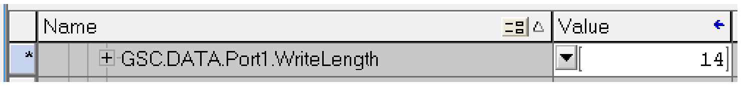
6.1 Sending ASCII Strings from the Processor to a Serial Device

1. In order to send data from the processor to a remote serial device, you must first copy the source data to the *GSC.DATA.PortX.WriteString* array tag:



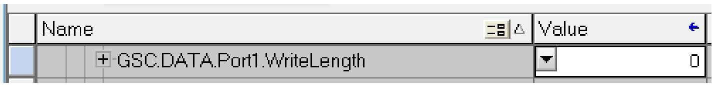
Name	Value
⊞ GSC.DATA.Port1.WriteLength	0
⊟ GSC.DATA.Port1.WriteString	{...}
⊞ GSC.DATA.Port1.WriteString[0]	'T'
⊞ GSC.DATA.Port1.WriteString[1]	'H'
⊞ GSC.DATA.Port1.WriteString[2]	'I'
⊞ GSC.DATA.Port1.WriteString[3]	'S'
⊞ GSC.DATA.Port1.WriteString[4]	' '
⊞ GSC.DATA.Port1.WriteString[5]	'I'
⊞ GSC.DATA.Port1.WriteString[6]	'S'
⊞ GSC.DATA.Port1.WriteString[7]	' '
⊞ GSC.DATA.Port1.WriteString[8]	'A'
⊞ GSC.DATA.Port1.WriteString[9]	' '
⊞ GSC.DATA.Port1.WriteString[10]	'T'
⊞ GSC.DATA.Port1.WriteString[11]	'E'
⊞ GSC.DATA.Port1.WriteString[12]	'S'
⊞ GSC.DATA.Port1.WriteString[13]	'T'

2. In the *GSC.DATA.PortX.WriteLength* tag, enter the number of characters to send. Once the AOI detects a non-zero value in the tag, the string data is sent to the remote serial device.



Name	Value
⊞ GSC.DATA.Port1.WriteLength	14

3. Once the operation is concluded, the AOI will automatically reset the WriteLength tag to '**0**'.



Name	Value
⊞ GSC.DATA.Port1.WriteLength	0

4. String diagnostics.
You can check the status of the data transmission by monitoring the *GSC.STATUS.Port1.TxMsgCnt* (transmitted message count) and *GSC.STATUS.Port1.TxCharCnt* (transmitted character count) tags.

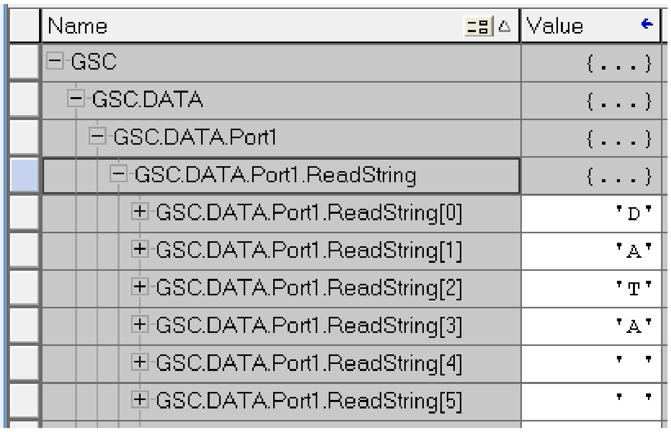
6.2 Receiving ASCII Strings from a Serial Device to the Processor

The MVI69E-GSC continuously buffers the data received from remote serial devices until the criteria to flush the data to the processor is met. The criteria to interrupt data buffering and flush the data to processor is determined by the *Type* configuration parameter which includes a combination of the following options:

- Termination Characters
- Message Timeout
- Intercharacter Delay
- Packet Size

For further information about the Type configuration parameter, refer to section ***GSC Port x Parameters***

The serial data received on the port is copied to the *GSC.DATA.PortX ReadString* tag.



Name	Value
⊟ GSC	{...}
⊟ GSC.DATA	{...}
⊟ GSC.DATA.Port1	{...}
⊟ GSC.DATA.Port1.ReadString	{...}
⊞ GSC.DATA.Port1.ReadString[0]	'D'
⊞ GSC.DATA.Port1.ReadString[1]	'A'
⊞ GSC.DATA.Port1.ReadString[2]	'T'
⊞ GSC.DATA.Port1.ReadString[3]	'A'
⊞ GSC.DATA.Port1.ReadString[4]	' '
⊞ GSC.DATA.Port1.ReadString[5]	' '

Note: Monitor the *GSC.STATUS.PortX* tag to indicate an incoming string has been received on the port. Copy the string from the *GSC.DATA.PortX ReadString* array to another array since the *GSC.DATA.PortX ReadString* array will be overwritten by subsequent, incoming strings.

7 Diagnostics and Troubleshooting

The module provides information on diagnostics and troubleshooting in the following forms:

- LED status indicators on the front of the module provide general information on the module's status.
- Status data contained in the module can be viewed through the Configuration/Debug port, using the troubleshooting and diagnostic capabilities of *ProSoft Configuration Builder (PCB)*.
- Status data values can be transferred from the module to processor memory and can be monitored there manually or by customer-created logic. For details on Status Data values, see MVI69E-GSC Status Data Area.

7.1 LED Status Indicators

The LEDs indicate the module’s operating status.

ETH	CFG
P1	BP
P2	OK

LED	Status	Description
ETH	Green	Application is running and Ethernet is ready
	Off	Possible causes: • Network communication has not started yet • Physical ethernet connection is down • Ethernet communication is disabled as the system shuts down.
P1	Green	Data is being transferred
	Red	Not used
	Off	No serial communication activity
P2	Green	Data is being transferred
	Red	Not used
	Off	No serial communication activity
CFG	Red	Possible causes: • Configuration failure during start-up or module boot • Module detected a bad or unreadable configuration file; problem opening the configuration file or the file is corrupted
	Green	Configuration is valid
	Amber	Possible causes: • During initialization, the configuration process has started and is being validated • Configuration file has been received and is being processed
	Off	Application is not running, or backplane has failed. Possible causes: • Communication with the backplane has failed • Backplane process is not up and running; no link to the controller so configuration status cannot be confirmed • Nothing to configure; module is waiting or idle. Cannot open file for writing or if no configuration file is received from the processor • Backplane data failure counter (*Backplane Fail Count* parameter in the Module configuration) exceeded the limit

		• Module is in shutdown function; configuration is no longer active once the system is shutting down.
BP	Red	Possible causes: • Backplane disruption or communication error (PLC may not be in RUN mode) • Initialization failed; backplane communication took too long. Block timeout occurred during module initialization • Backplane data failure counter (*Backplane Fail Count* parameter in the Module configuration) has exceeded the limit.
	Green	Backplane transfers are successful; communications with PLC are operational
	Amber	Initialization state: Module is in the process of establishing communication with the controller.
	Off	Possible causes: • During power-up, no communication with the backplane has started yet • Module is in shutdown function; backplane link is no longer active once the system is shutting down.
OK	Red	Possible causes: • Module is powering up • Module has encountered a fatal error • Module is in shutdown function
	Green	The module has properly initialized and is running

If the BP ACT and OK LEDs blink at a rate of every one-second, this indicates a serious problem with the module. Call ProSoft Technology support to arrange for repairs.

7.2 Ethernet LED Indicators

The Ethernet LEDs indicate the module's Ethernet port status.

LED	State	Description
100 Mbit	Off	Ethernet connected at 10Mbps duplex speed
	Amber Solid	Ethernet connected at 100Mbps duplex speed
LINK/ACT	Off	No physical network connection is detected. No Ethernet communication is possible. Check wiring and cables.
	Green Solid or Blinking	Physical network connection detected. This LED must be ON solid for Ethernet communication to be possible.

7.3 Clearing a Fault Condition

Typically, if the OK LED turns red for more than ten seconds, a hardware problem has been detected in the module or the program has exited.

To clear the condition, follow these steps:

1 Turn off power to the rack.
2 Remove the card from the rack.
3 Verify that all jumpers are set correctly.
4 Re-insert the card in the rack and turn the power back on.
5 Verify correct configuration data is being transferred to the module from the CompactLogix or MicroLogix controller.

If the module's OK LED does not turn green, verify that the module is inserted completely into the rack. If this does not resolve the problem, contact ProSoft Technology Technical Support.

7.4 Troubleshooting

Use the following troubleshooting steps if you encounter problems when the module is powered up. If these steps do not resolve your problem, please contact ProSoft Technology Technical Support.

7.4.1 Processor Errors

Problem description	Steps to take
Processor fault	Verify that the module is securely plugged into the slot that has been configured for the module in the I/O Configuration in RSLogix. Verify that the slot location in the rack has been configured correctly in the ladder logic.
Processor I/O LED flashes	This indicates a problem with backplane communications. A problem could exist between the processor and any installed I/O module, not just the MVI69E-GSC. Verify that all modules in the rack are correctly configured.

7.4.2 Module Errors

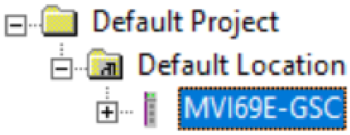
Problem description	Steps to take
BP ACT LED remains Off or blinks slowly	This indicates that backplane transfer operations are failing. To establish backplane communications, verify the following items: ▪ The processor is in RUN or REM RUN mode. ▪ The backplane driver is loaded in the module. ▪ The module is configured for read and write data block transfer. ▪ The ladder logic handles all read and write block situations. ▪ The module is properly configured in the processor I/O configuration and ladder logic.
OK LED remains red	The program has halted or a critical error has occurred. If the program has halted, turn off power to the rack, remove the card from the rack, then re-insert it, and then restore power to the rack.

7.5 Configuring a Temporary IP Address

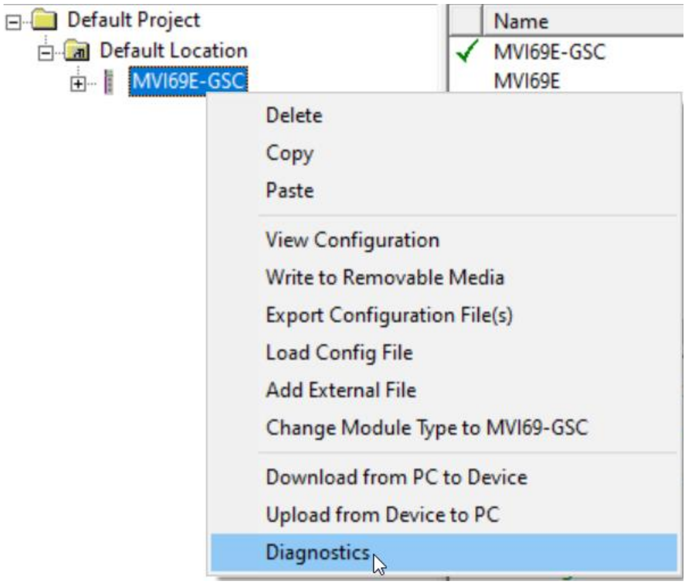
This feature is useful if your laptop is on a different subnet than the module's IP address. The module can be configured to use the same subnet as your laptop.

> **Important: ProSoft Configuration Builder** (PCB) locates MVI69E-GSC modules through UDP broadcast messages. These messages may be blocked by routers or layer 3 switches. In that case, **ProSoft Discovery Service** is unable to locate the modules.
>
> To use **PCB**, arrange the Ethernet connection so that there is no router/ layer 3 switch between the computer and the module, OR reconfigure the router/ layer 3 switch to allow routing of the UDP broadcast messages.

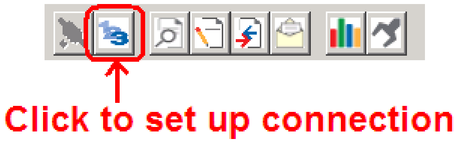
1 In the tree view in **ProSoft Configuration Builder (PCB)**, select the **MVI69E-GSC** module.



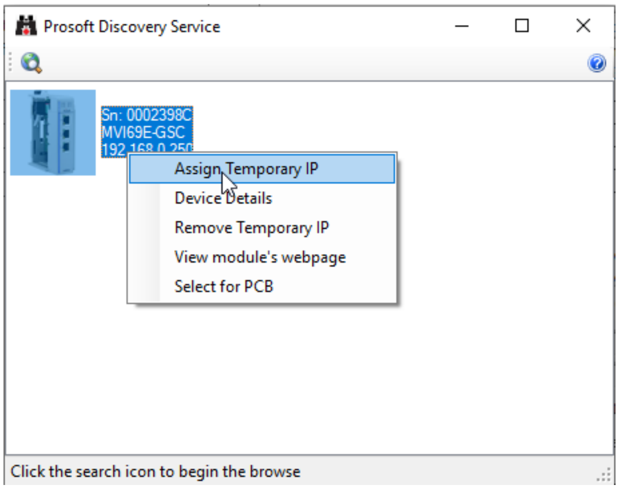
2 Right-click the module icon in the tree and choose **DIAGNOSTICS.**



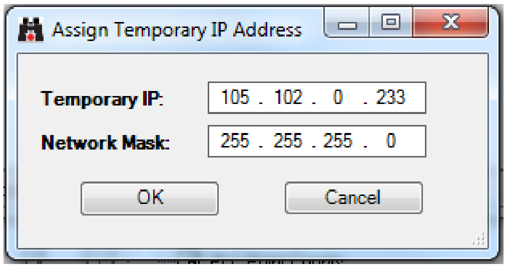
3 In the *Diagnostics* window, click the **SET UP CONNECTION** button.



4 In the *Connection Setup* dialog box, click **BROWSE DEVICE(S)** to start *ProSoft Discovery Service***.** Right-click the module and choose **ASSIGN TEMPORARY IP**.



5 The module's default IP address is usually 192.168.0.250. Choose an unused IP within your subnet, and then click **OK**.



Temporary IP: 105 . 102 . 0 . 233

Network Mask: 255 . 255 . 255 . 0

Important: The temporary IP address is only valid until the next time the module is initialized.

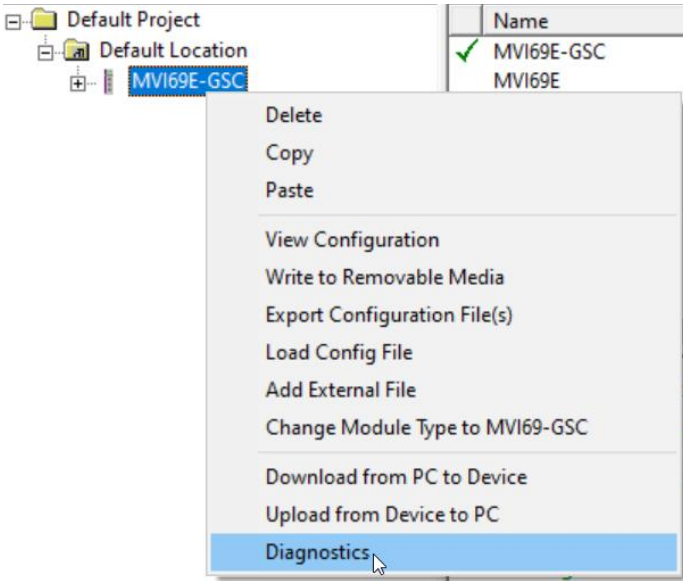
6 Close the *ProSoft Discovery Service* window. Enter the temporary IP address in the **ETHERNET ADDRESS** field of the *Connection Setup* dialog box, then click **TEST CONNECTION** to verify that the module is accessible with the current settings.

7 If the *Test Connection* is successful, click **CONNECT**. The *Diagnostics* window is now accessible.

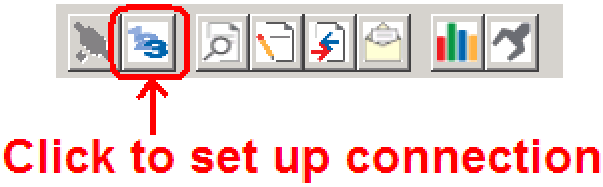
ProSoft Configuration Builder (PCB) provides diagnostic menus for debugging and troubleshooting.

7.6 Diagnostics Menu

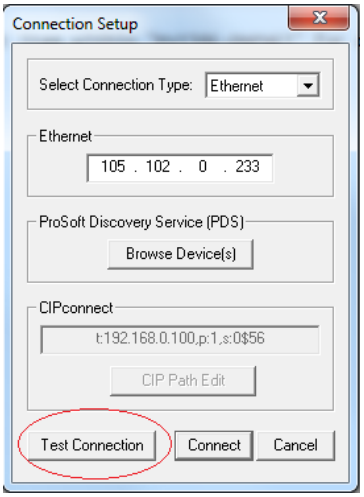
1 In the tree view in ProSoft Configuration Builder, right-click the **MVI69E-GSC** module and then choose **DIAGNOSTICS**.



2 After the *Diagnostics* window opens, click the **SET UP CONNECTION** button to browse for the module's IP address.



3 In the *Ethernet* field of the *Connection Setup* dialog box, enter the current IP address, whether it is temporary or permanent. Click **TEST CONNECTION** to verify that the module is accessible with the current settings.

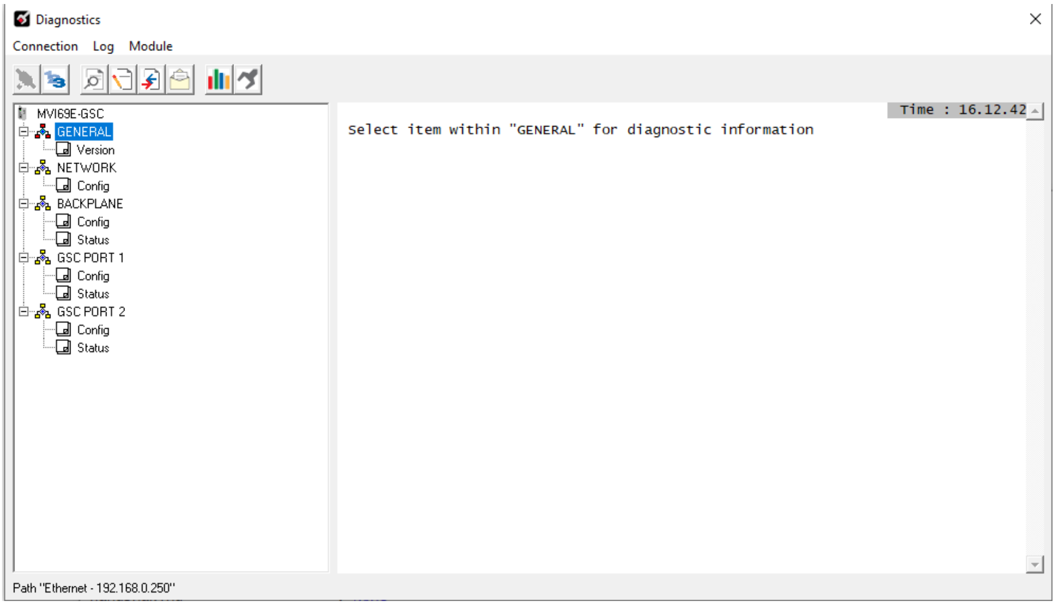


Select Connection Type: Ethernet

Ethernet: 105 . 102 . 0 . 233

CIPconnect: t:192.168.0.100,p:1,s:0$56

4 If the **TEST CONNECTION** is successful, click **CONNECT**. The *Diagnostics* window is now visible.

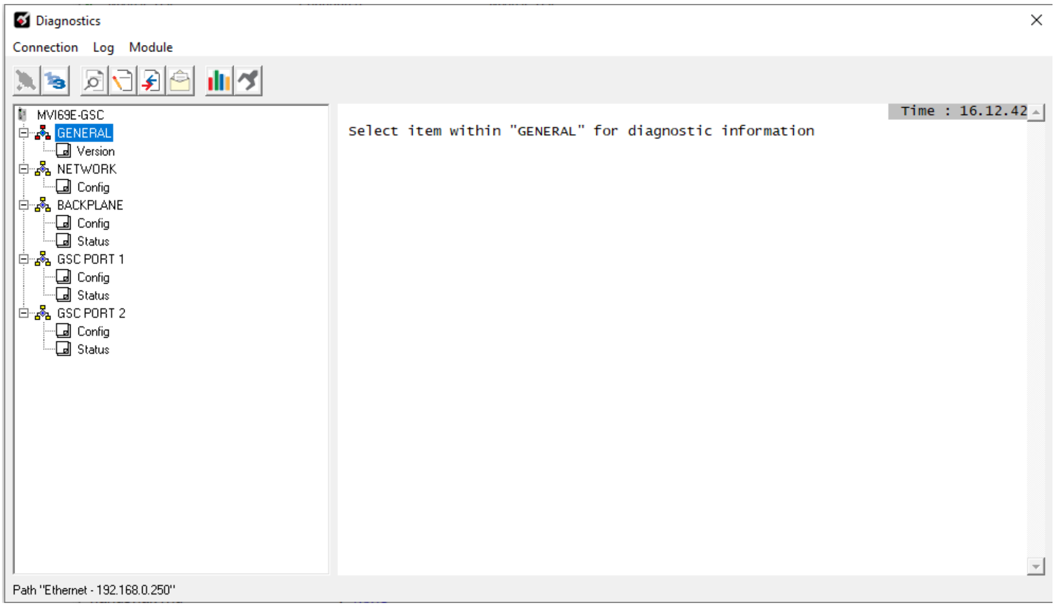


```
Time : 16.12.42
Select item within "GENERAL" for diagnostic information
```

Path "Ethernet - 192.168.0.250"

7.6.1 Diagnostics Menu Navigation

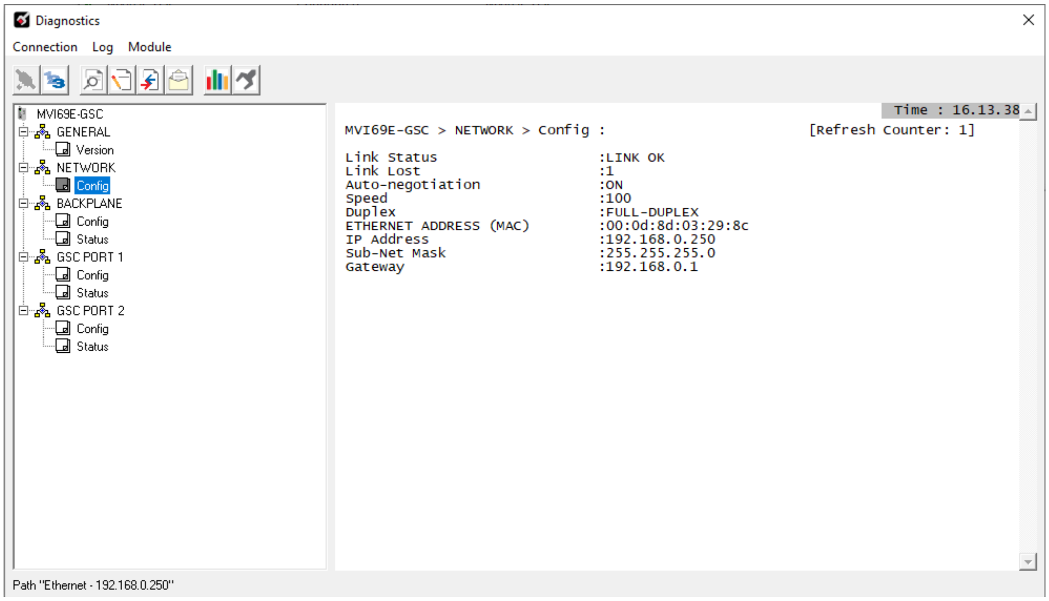
In the *Diagnostics* window in ProSoft Configuration Builder, the Diagnostics menu is available through the Ethernet configuration port. The menu is arranged as a tree structure.



```
Time : 16.12.42
Select item within "GENERAL" for diagnostic information
```

7.6.2 Monitoring Network Configuration Information

In the *Diagnostics* window in Prosoft Configuration Builder, click **NETWORK** and then click **CONFIG** to view the Ethernet network configuration information.

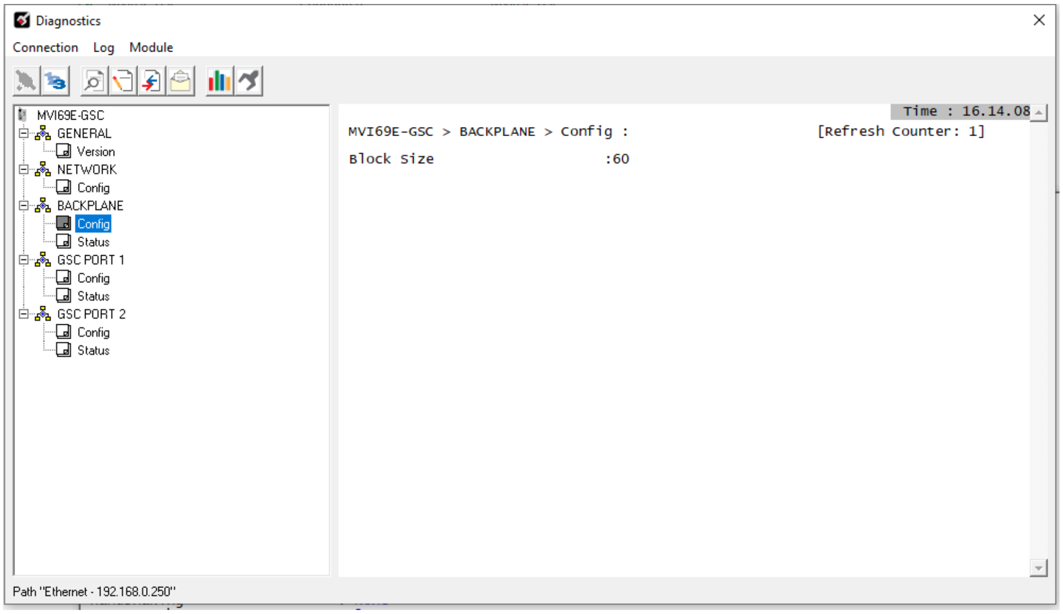


```
                                                              Time : 16.13.38
MVI69E-GSC > NETWORK > Config :                            [Refresh Counter: 1]

Link Status                         :LINK OK
Link Lost                           :1
Auto-negotiation                    :ON
Speed                               :100
Duplex                              :FULL-DUPLEX
ETHERNET ADDRESS (MAC)              :00:0d:8d:03:29:8c
IP Address                          :192.168.0.250
Sub-Net Mask                        :255.255.255.0
Gateway                             :192.168.0.1
```

7.6.3 Monitoring Backplane Information

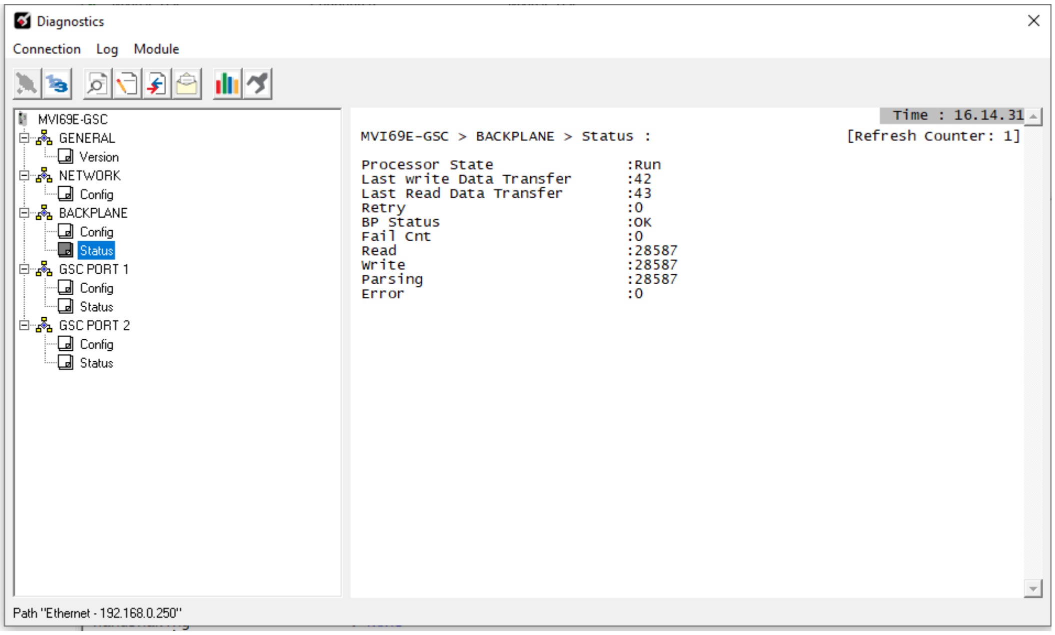
In the *Diagnostics* window in ProSoft Configuration Builder, click **BACKPLANE** to view the backplane information. This menu has two submenus:

- **CONFIGURATION**
- **STATUS**



```
MVI69E-GSC > BACKPLANE > Config :                    [Refresh Counter: 1]

Block Size                      :60
```

Path "Ethernet - 192.168.0.250"



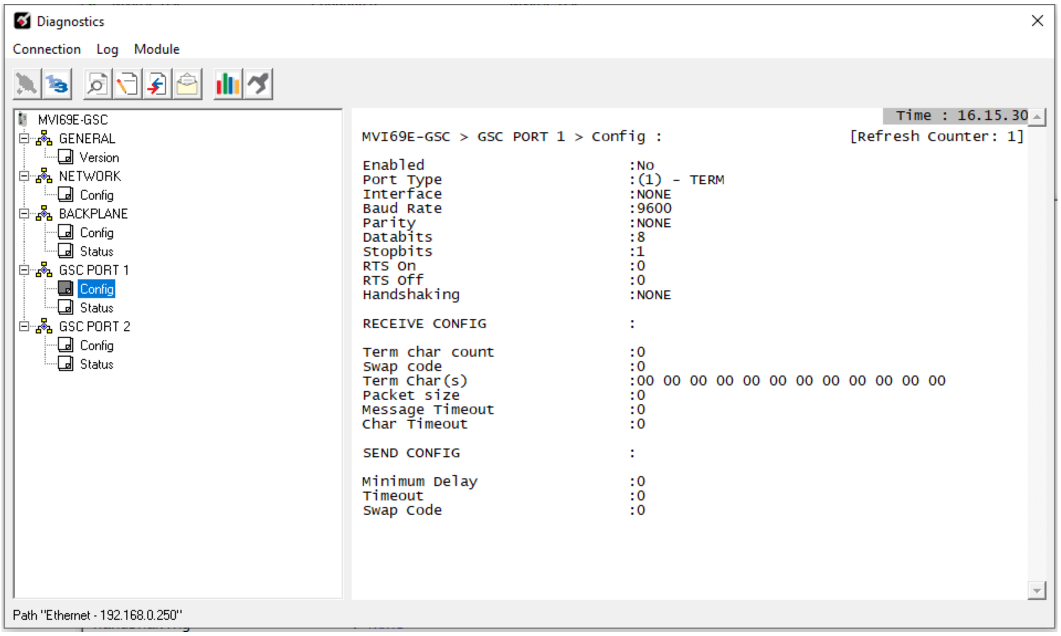
```
MVI69E-GSC > BACKPLANE > Status :                    [Refresh Counter: 1]

Processor State             :Run
Last Write Data Transfer    :42
Last Read Data Transfer     :43
Retry                       :0
BP Status                   :OK
Fail Cnt                    :0
Read                        :28587
Write                       :28587
Parsing                     :28587
Error                       :0
```

Path "Ethernet - 192.168.0.250"

7.6.4 Port x Module Information

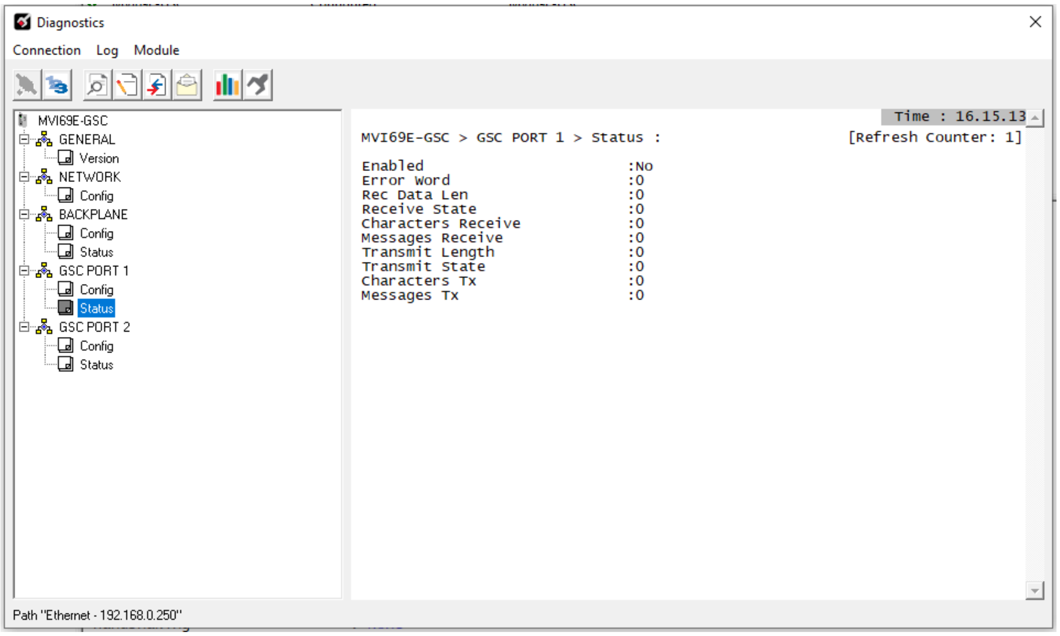
The **GSC Port 1** and **GSC Port 2** menus include the following submenus:

- Configuration
- Status (General status for the port)



```
MVI69E-GSC > GSC PORT 1 > Config :                    [Refresh Counter: 1]

Enabled                        :No
Port Type                      :(1) - TERM
Interface                      :NONE
Baud Rate                      :9600
Parity                         :NONE
Databits                       :8
Stopbits                       :1
RTS On                         :0
RTS Off                        :0
Handshaking                    :NONE

RECEIVE CONFIG                 :

Term char count                :0
Swap code                      :0
Term Char(s)                   :00 00 00 00 00 00 00 00 00 00 00 00
Packet size                    :0
Message Timeout                :0
Char Timeout                   :0

SEND CONFIG                    :

Minimum Delay                  :0
Timeout                        :0
Swap Code                      :0
```

Path "Ethernet - 192.168.0.250"



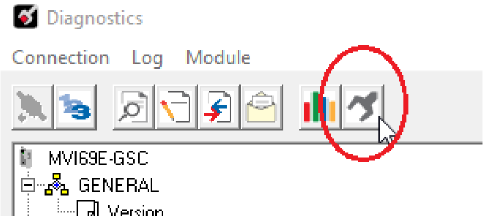
```
MVI69E-GSC > GSC PORT 1 > Status :                    [Refresh Counter: 1]

Enabled                        :No
Error Word                     :0
Rec Data Len                   :0
Receive State                  :0
Characters Receive             :0
Messages Receive               :0
Transmit Length                :0
Transmit State                 :0
Characters Tx                  :0
Messages Tx                    :0
```

Path "Ethernet - 192.168.0.250"

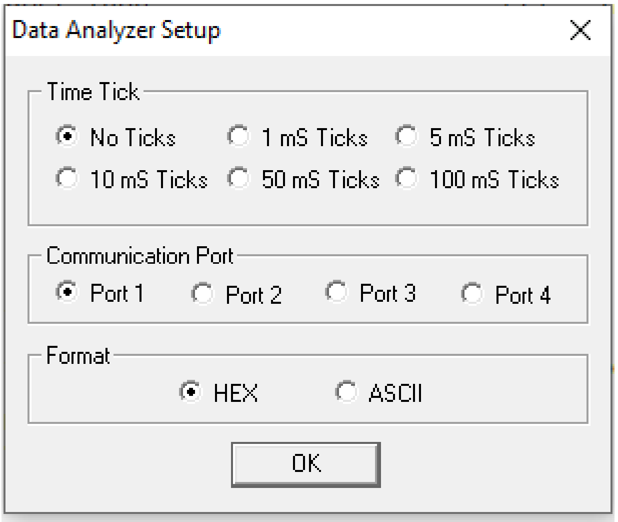
7.6.5 Data Analyzer

The Data Analyzer feature allows you to view the serial communications traffic as it is received and sent from the selected MVI69E-GSC communications port.

1 Click on the **SETUP DATA ANALYZER** icon.



2 Select the *Time Tick* setting, the port number, and the data format.



Data Analyzer Setup

Time Tick

● No Ticks ○ 1 mS Ticks ○ 5 mS Ticks
○ 10 mS Ticks ○ 50 mS Ticks ○ 100 mS Ticks

Communication Port

● Port 1 ○ Port 2 ○ Port 3 ○ Port 4

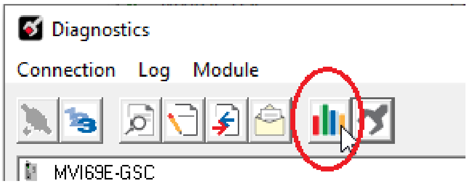
Format

● HEX ○ ASCII

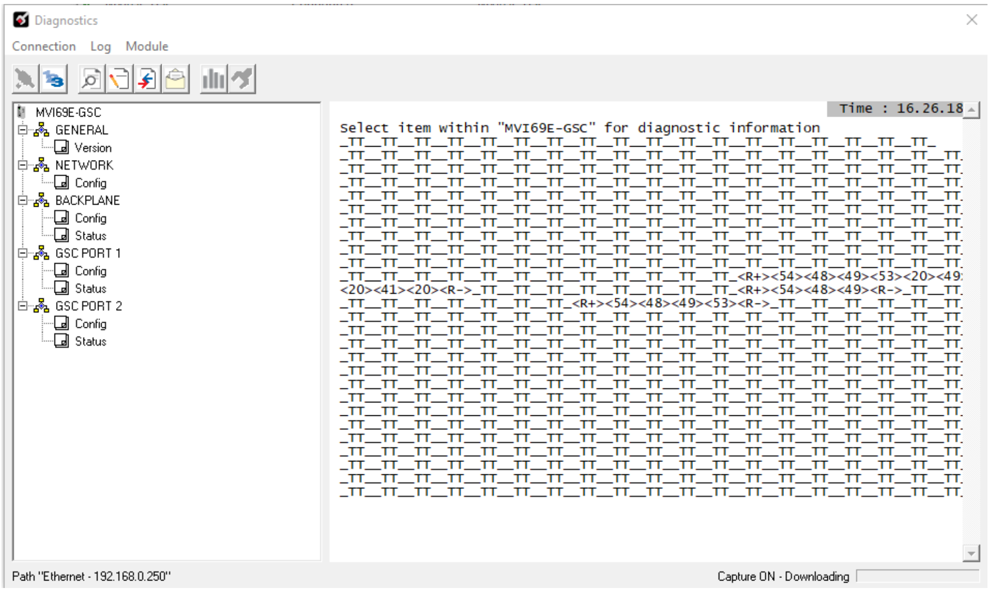
OK

3 Click **OK** to confirm.

4 Select the **START DATA ANALYZER** button to start the data analyzer capture.



5 Each byte will be enclosed with < > for data transferred out of the port, and [ ] for data received on the port.

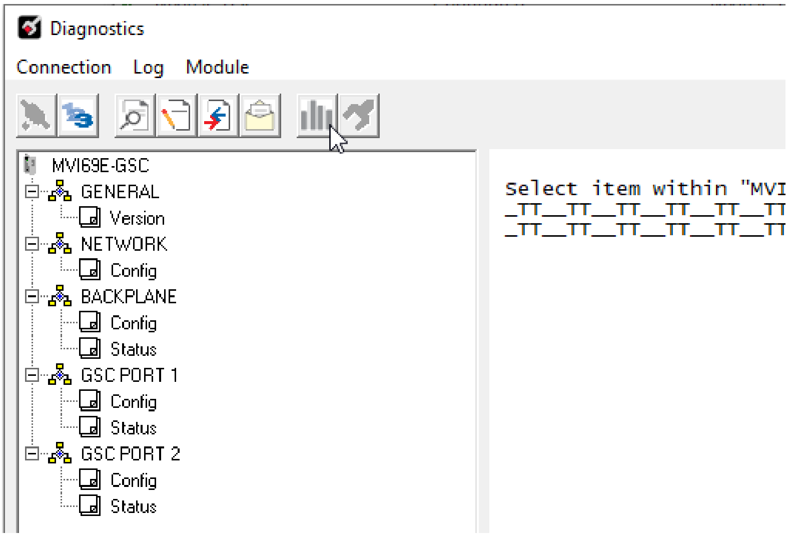


```
Time : 16.26.18
Select item within "MVI69E-GSC" for diagnostic information
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_<R+><54><48><49><53><20><49
<20><41><20><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><54><48><49><R->_TT__TT_
_TT__TT__TT__TT__TT__TT__TT_<R+><54><48><49><53><R->_TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
_TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT__TT_
```

Path "Ethernet - 192.168.0.250"

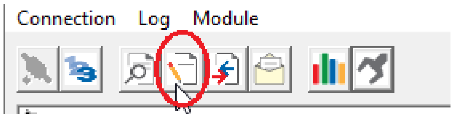
Capture ON - Downloading

6 Click on the same **DATA ANALYZER** button to stop the data analyzer operation:

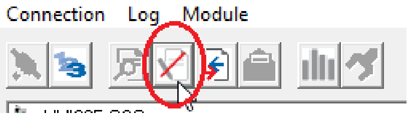


```
Select item within "MVI
_TT__TT__TT__TT__TT__TT
_TT__TT__TT__TT__TT__TT
```

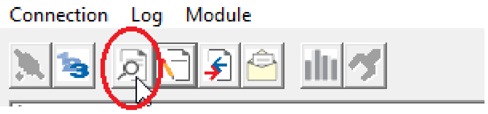
7 You may also log the data analyzer capture to a text file. Click on the following **START LOG** button to log the data analyzer capture:



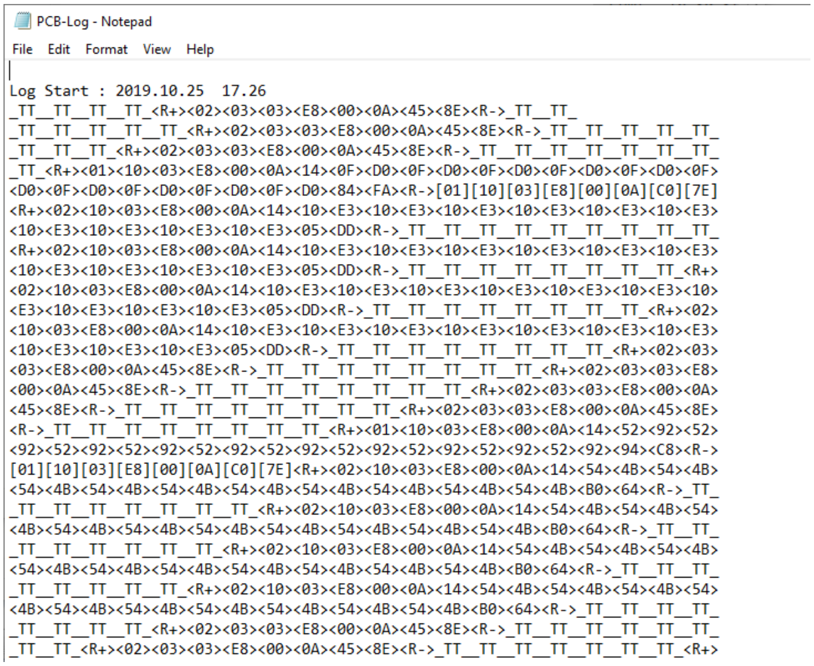
8 Click on the same **LOG** button to stop the data capture logging:



9 Click on the **VIEW LOG FILE** button to view the log file:

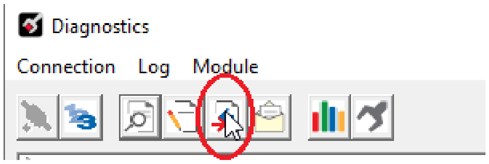


10 The log file opens in Notepad



```
Log Start : 2019.10.25  17.26
_TT__TT__TT__TT_<R+><02><03><03><E8><00><0A><45><8E><R->_TT__TT_
_TT__TT__TT__TT__TT_<R+><02><03><03><E8><00><0A><45><8E><R->_TT__TT__TT__TT__TT_
_TT__TT__TT_<R+><02><03><03><E8><00><0A><45><8E><R->_TT__TT__TT__TT__TT__TT__TT_
_TT_<R+><01><10><03><E8><00><0A><14><0F><D0><0F><D0><0F><D0><0F><D0><0F><D0><0F>
<D0><0F><D0><0F><D0><0F><D0><0F><D0><84><FA><R->[01][10][03][E8][00][0A][C0][7E]
<R+><02><10><03><E8><00><0A><14><10><E3><10><E3><10><E3><10><E3><10><E3><10><E3>
<10><E3><10><E3><10><E3><10><E3><05><DD><R->_TT__TT__TT__TT__TT__TT__TT__TT__TT_
<R+><02><10><03><E8><00><0A><14><10><E3><10><E3><10><E3><10><E3><10><E3><10><E3>
<10><E3><10><E3><10><E3><10><E3><05><DD><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+>
<02><10><03><E8><00><0A><14><10><E3><10><E3><10><E3><10><E3><10><E3><10><E3><10>
<E3><10><E3><10><E3><10><E3><05><DD><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><02>
<10><03><E8><00><0A><14><10><E3><10><E3><10><E3><10><E3><10><E3><10><E3><10><E3>
<10><E3><10><E3><10><E3><05><DD><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><02><03>
<03><E8><00><0A><45><8E><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><02><03><03><E8>
<00><0A><45><8E><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><02><03><03><E8><00><0A>
<45><8E><R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><02><03><03><E8><00><0A><45><8E>
<R->_TT__TT__TT__TT__TT__TT__TT__TT_<R+><01><10><03><E8><00><0A><14><52><92><52>
<92><52><92><52><92><52><92><52><92><52><92><52><92><52><92><52><92><94><C8><R->
[01][10][03][E8][00][0A][C0][7E]<R+><02><10><03><E8><00><0A><14><54><4B><54><4B>
<54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><B0><64><R->_TT_
_TT__TT__TT__TT__TT__TT__TT_<R+><02><10><03><E8><00><0A><14><54><4B><54><4B><54>
<4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><B0><64><R->_TT__TT_
_TT__TT__TT__TT__TT__TT_<R+><02><10><03><E8><00><0A><14><54><4B><54><4B><54><4B>
<54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><B0><64><R->_TT__TT__TT_
_TT__TT__TT__TT__TT_<R+><02><10><03><E8><00><0A><14><54><4B><54><4B><54><4B><54>
<4B><54><4B><54><4B><54><4B><54><4B><54><4B><54><4B><B0><64><R->_TT__TT__TT__TT_
_TT__TT__TT__TT_<R+><02><03><03><E8><00><0A><45><8E><R->_TT__TT__TT__TT__TT__TT_
_TT__TT_<R+><02><03><03><E8><00><0A><45><8E><R->_TT__TT__TT__TT__TT__TT__TT_<R+>
```

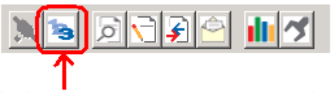
11 To clear the log file before a new logging session starts, click the **CLEAR LOG FILE** button.



7.7 Connecting to the Module's Webpage

The module's internal web server provides access to module version and status information, as well as the ability to set the date and time, reboot the module, and download firmware upgrade to the module. Enter the assigned IP address of the module into a web browser or use the following steps in PCB.

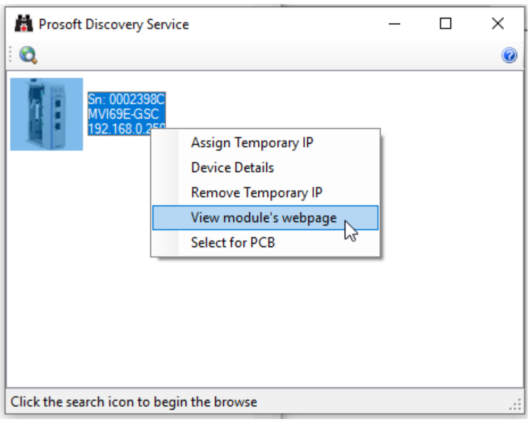
1 In the PCB Diagnostics window, click the **SET UP CONNECTION** button.



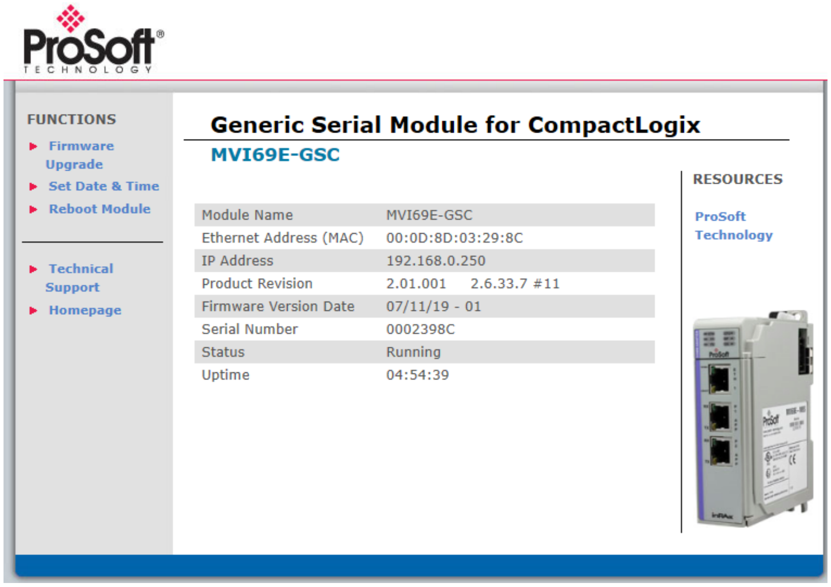
Click to set up connection

2 In the *Connection Setup* dialog box, click **BROWSE DEVICE(S)** to start *ProSoft Discovery Service***.**

3 Right-click the module icon and choose **VIEW MODULE'S WEBPAGE** to launch your default browser and display the module's webpage.



Click the search icon to begin the browse



Generic Serial Module for CompactLogix

MVI69E-GSC

Module Name	MVI69E-GSC
Ethernet Address (MAC)	00:0D:8D:03:29:8C
IP Address	192.168.0.250
Product Revision	2.01.001 2.6.33.7 #11
Firmware Version Date	07/11/19 - 01
Serial Number	0002398C
Status	Running
Uptime	04:54:39

8 Reference

8.1 Product Specifications

The MVI69E-GSC allows Rockwell Automation® CompactLogix or MicroLogix 1500-LRP® I/O compatible processors to interface easily with other ASCII protocol compatible devices.

The module acts as an input/output communications module between the ASCII network and the CompactLogix or MicroLogix 1500-LRP backplane. The data transfer from the CompactLogix or MicroLogix 1500-LRP processor is asynchronous from the actions on the Modbus network. Databases are user-defined and stored in the module to hold the data required by the protocol.

- Single-slot, 1769 backplane-compatible
- The module is recognized as an Input/Output module and has access to processor memory for data transfer between processor and module.
- Ladder Logic is used for data transfer between module and processor. Sample Add-On Instruction file included.
- Configuration data obtained from and stored in the module
- Supports CompactLogix or MicroLogix 1500-LRP processors with 1769 I/O bus capability and at least 500 mA of 5 VDC backplane current available.

8.1.1 Hardware Specifications

Specification	Description
Dimensions	Standard 1769 Single-slot module
Current Load	500 mA max @ 5 VDC Power supply distance rating of 4 (L43 and L45 installations on first 4 slots of 1769 bus)
Operating Temp.	32° F to 140° F (0° C to 60°C)
Storage Temp.	-40° F to 185° F (-40° C to 85° C)
Relative Humidity	5% to 95% (with no condensation)
LED Indicators	Module OK Status Backplane Activity Ethernet Port Activity Configuration Activity Serial port Activity
CFG Port (ETH)	Diagnostics over Ethernet connection
App Ports (P1,P2)	RS-232, RS-485 or RS-422 (jumper selectable) RJ45 Port (DB-9F with supplied cable) 500V Optical isolation from backplane
Shipped with Unit	RJ45 to DB-9M cables for each application port

8.1.2 Functional Specifications

- ASCII Communication ports 1 & 2 (PRT1, PRT2)
  - Both ports are capable of transmitting and/or receiving ASCII character strings. Each port is individually configurable:
  - Termination types
  - Stream mode
  - Termination character(s)
  - Message timeout
  - Intercharacter timeout
  - Packet size limit
  - Baud rate: 110 to 115.2K baud
  - Parity: none, even, odd
  - Stop bits: 1 or 2
  - Data bits: 5 to 8
  - RTS on/off timing: 0 to 65535 milliseconds
  - Minimum response delay: 0 to 65535 milliseconds
- Handshaking (optional)
  - Hardware: RTS/CTS, DTR/DSR
  - Software: XON/XOFF
- ASCII character strings up to 4096 characters in length supported
- Full hardware handshaking control provides radio, modem and multi-drop support
- User-definable module memory usage, supporting the storage and transfer of up to 4000 registers to/from the control processor
- Module error and status conditions returned to processor for diagnostic purposes
  - Module status
  - Port error status word (bit mapped)
  - Port receive state
  - Port receive character count
  - Port receive block count
  - Port transmit state
  - Port transmit character count
  - Port transmit block count
- All data related to the module is contained in a single controller tag with defined objects to simplify configuration, monitoring and interfacing with the module
- Module configuration and communication configuration data is transferred to the module via a predefined user data type in the processor

8.2 Functional Overview

8.2.1 General Concepts

The MVI69E-GSC module uses ladder logic to communicate with the CompactLogix or MicroLogix 1500-LRP processor across the backplane. The ladder logic handles the module data transfer, configuration data transfer, special block handling, and status data receipt.

The following topics describe several concepts that are important for understanding the operation of the MVI69E-GSC module. This is the order of operations on power-up:

1. The module begins the following logical functions:
   - Initialize hardware components
   - Initialize CompactLogix or MicroLogix 1500-LRP backplane driver
   - Test and clear all RAM
2. Read configuration from the CompactLogix or MicroLogix 1500-LRP processor through ladder logic
3. Allocate and initialize Module Register space
4. Enable application port(s)

After the module has received the module configuration, the module begins communicating with other devices on the network, depending on the configuration of the module.

8.2.2 Backplane Data Flow

The following topics describe the flow of data between the hardware (CompactLogix processor and MVI69E-GSC module) and other devices on the network under the module’s different operating modes.

Backplane Data Transfer

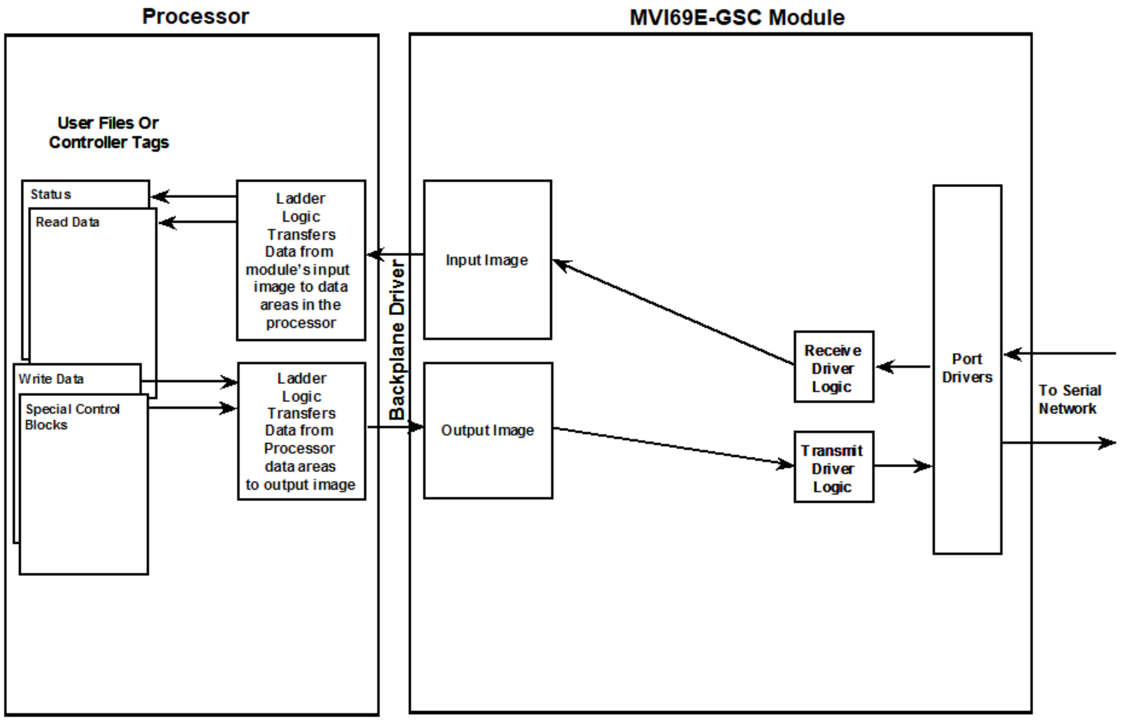
The MVI69E-GSC module communicates directly over the CompactLogix or MicroLogix backplane. Data is paged between the module and the CompactLogix processor across the backplane using the module's input and output images. The update frequency of the images is determined by the scheduled scan rate defined by the user for the module and the communication load on the module. Typical updates are in the range of 2 to 10 milliseconds.

The data is paged between the processor and the module using input and output image blocks (fixed at 60 words).

This bi-directional transference of data is accomplished by the module filling in data in the module's input image to send to the processor. Data in the input image is placed in the Controller Tags in the processor by the ladder logic.

The processor inserts data to the module's output image to transfer to the module. The module's program extracts the data and places it in the module's internal database.

The following illustration shows the data transfer method used to move data between the CompactLogix processor, the MVI69E-GSC module, and the network.


Processor
MVI69E-GSC Module
User Files Or Controller Tags
Status
Read Data
Write Data
Special Control Blocks
Ladder Logic Transfers Data from module's input image to data areas in the processor
Ladder Logic Transfers Data from Processor data areas to output image
Backplane Driver
Input Image
Output Image
Receive Driver Logic
Transmit Driver Logic
Port Drivers
To Serial Network

Normal Data Transfer

Normal data transfer includes the paging of the user data to and from the module's ports and bringing in status data. These data are transferred through read (input image) and write (output image) blocks. The structure and function of each block is discussed in the following topics.

Block Request from the Processor to the Module

These blocks of data are used to transfer information from the CompactLogix processor to the module. The structure of the output image used to transfer this data is shown below:

Port	Word Offset	Description
Port 1	0	Block Sequence Number (Read block number as sent by module) (0 to 127)
	1	Inter-character delay for this message (milliseconds between characters)
	2	Number of characters to transmit on Port 1 (0 to 50)
	3 to 27	Port 1 ASCII character codes to transmit (up to 50 ASCII characters)
Port 2	28	Inter-character delay for this message (milliseconds between characters)
	29	Number of characters to transmit on Port 2 (0 to 50)
	30 to 54	Port 2 ASCII character codes to transmit (up to 50 ASCII characters)
	55 to 59	Reserved

The Block Sequence Number is that received on the last read block transfer through the input image on the module. The ladder logic should copy this value from byte 0 of the input image to byte 0 of output image in the ladder logic. This must be the last operation performed when constructing the write block. The module's program will trigger the process write block function when a new value is recognized in byte 0 of the output image.

If the number of characters to transmit in the write block is not set to zero (non-zero value in bytes 2 and 29), this indicates to the module there is data present in the block that needs to be transmitted. If the selected port is not already busy transmitting data from a previous write block, the data in the block will be moved to the port's transmit buffer and sent out the port as soon as possible.

In order to pace the characters for the write operation, an inter-character delay value is associated with each write message. For devices that do not buffer received data, when interfacing with a modem in command mode or when simulating keyboard or keypad entry, inter-character delays may be required. For example, if the port is tied to a device that expects input with delays of 200 milliseconds between each character, place the data to send in the write block output image along with the length and set the inter-character delay byte (bytes 1 and 28) to a value of 200 in the module's output image in the processor's ladder logic program.

The message will be transmitted with a 200-millisecond wait period between each character. Because this delay value is sent from the processor for each write message, the inter-character delay can be set independently for each message. For example, when writing AT commands to a dial-up modem, an inter-character delay of 100 may be required. But when the modem is in data mode, the inter-character delay can be set to 0. When the delay is set to 0, the whole packet of data will be placed in the module's transmit buffer at one time.

Block Response from the Module to the Processor

These blocks transfer information from the module to the processor. The structure of the input image used to transfer these data is shown below. The Block Sequence Number (byte 0) is an index value used to signal to the processor that a new block is ready for processing. The ladder logic must recognize a change in this value and process the data encapsulated in the input image. The block contains the data received on each port and status data. The two byte values in bytes 1 (port 1 receive length), and 28 (port 2 receive length), hold the number of characters received on each port to be processed by the ladder logic. ASCII character code data received on the ports are found starting at byte 3 and 30 for Port 1 and 2, respectively. The simpler version of the example ladder logic assumes the number of ASCII characters received on each port is less than or equal to fifty characters (25 words per port, times 2 characters per word).

	Word Offset	Description
	0	Block Sequence Number (Bumped each scan by module) (0 to 127)
Port 1	1	Number of characters (0 to 50) in Port 1 receive block (3 to 27). If the string received on the port is larger than 50 characters, multiple blocks will be transferred. Any block with a value of -1 in this field represents the first or continuation block and the block contains 50 characters of ASCII code data. The last block of data will contain a positive number in this field that represents the number of characters in the last block. Status data will be returned in words 3 to 27 if this word contains a value of 0.
	2	Number of characters transmitted (0 to 50) from last block write for Port 1
	3 to 27	Port 1 data received (up to 50 ASCII character codes of data). If the number of characters received for the port is 0, status data will be returned in this area.
Port 2	28	Number of characters (0 to 50) in Port 2 receive block (30 to 54). If the string received on the port is larger than 50 bytes, multiple blocks will be transferred. Any block with a value of -1 in this field represents the first of continuation block and the block contains 50 characters of ASCII code data. The last block of data will contain a positive number in this field that represents the number of characters in the last block. Status data will be returned in words 30 to 54 if this word contains a value of 0.
	29	Number of characters transmitted (0 to 50) from last block write for Port 2
	30 to 54	Port 2 data received (up to 50 ASCII character codes of data). If the number of characters received for the port is 0, status data will be returned in this area
	55 to 59	Reserved

The receive buffer in the module can hold up to 4096 characters. This large size permits the buffering of a large amount of data before a transfer of the data to the controller is required. The module buffers incoming ASCII characters in its receive buffer until one of the user-specified termination conditions is recognized. The module will then transfer the received terminated string to the controller.

The ladder logic required to properly handle transfer of terminated strings longer than 50 characters per port is more complex than the simpler version of ladder logic discussed above. If the terminated string is larger than 50 characters, multiple blocks will be used to transfer the data to the controller. The first block will contain a value of -1 in the "Number of Characters Received" data field. This indicates that there will be more blocks to follow and that the current block contains 50 ASCII character codes. As long as more than 50 characters remain in the buffer waiting to be sent to the ladder logic, successive Read Blocks will continue to show the "Number of Characters Received" as -1. When 50 or fewer ASCII characters remain in the buffer, the module will send the last block with a positive number in the length field. The value passed represents the number of characters present in the data area, which is the last characters of the complete, terminated string. The ladder logic must recognize the presence of one or more successive blocks with -1 lengths and then the positive number of the last block as indication that a single, complete, long string has been completely transferred.

The two byte values at bytes 1 (port 1 transmit count) and 29 (port 2 transmit count) inform the processor of the number of ASCII characters transferred in the last write block to the respective port transmit buffers. If a value of zero is returned in one of these words and data was sent in the last write block, the ladder logic must re-send the data in the next write block because the port was in a busy state and could not transmit the last data to be written at the time the Write Block was receive by the module from the ladder logic. If a non-zero value is returned in one of these bytes, the value represents the number of ASCII characters from the last write block that were successfully moved into the port's transmit buffer.

The status information transferred in the read block can be used by the processor to determine the state and "health" of the module and the device(s) attached to each application port. An important member of the value in the status object is error word for each port. This value contains the configuration error flags for each port and the receive buffer overflow error flag.

8.2.3 Special Function Blocks

Special Function blocks are blocks used to control the module or request special data from the module. The current version of the software supports two Special Functions, warm boot and cold boot.

Warm Boot Block (9998)

This block is sent from the CompactLogix or MicroLogix processor to the module (output image) when the module is required to perform a warm-boot (software reset) operation. The following table describes the format of the control block.

Offset	Description	Length
0	9998	1
1	Spare	

Cold Boot Block (9999)

This block is sent from the CompactLogix or MicroLogix processor to the module (output image) when the module is required to perform the cold boot (hardware reset) operation. This block is sent to the module when a hardware problem is detected by the ladder logic that requires a hardware reset. The following table describes the format of the control block.

Offset	Description	Length
0	9999	1
1	Spare	

8.3 Ethernet Port Connection

8.3.1 Ethernet Cable Specifications

The recommended cable is Category 5 or better. A Category 5 cable has four twisted pairs of wires, which are color-coded and cannot be swapped. The module uses only two of the four pairs.

The Ethernet port or ports on the module are Auto-Sensing. You can use either a standard Ethernet straight-through cable or a crossover cable when connecting the module to an Ethernet hub, a 10/100 Base-T Ethernet switch, or directly to a PC. The module detects the cable type and uses the appropriate pins to send and receive Ethernet signals.

Some hubs have one input that can accept either a straight-through or crossover cable, depending on a switch position. In this case, you must ensure that the switch position and cable type agree.

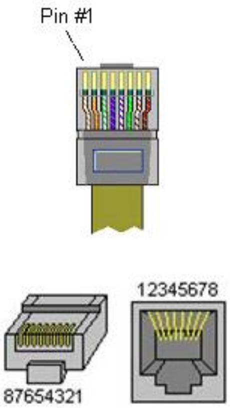
Refer to Ethernet Cable Configuration (page 70) for a diagram of how to configure Ethernet cable.

Ethernet Cable Configuration

Note: The standard connector view shown is color-coded for a straight-through cable.

Crossover cable

RJ-45 PIN	RJ-45 PIN
1 Rx+	3 Tx+
2 Rx-	6 Tx-
3 Tx+	1 Rx+
6 Tx-	2 Rx-



Straight- through cable

RJ-45 PIN	RJ-45 PIN
1 Rx+	1 Tx+
2 Rx-	2 Tx-
3 Tx+	3 Rx+
6 Tx-	6 Rx-

Ethernet Performance

Ethernet performance in the MVI69E-GSC module can be affected in the following way:

- Accessing the web interface (refreshing the page, downloading files, and so on) may affect performance
- Also, high Ethernet traffic may impact module performance, so consider one of these options:
    - Use managed switches to reduce traffic coming to module port
    - Use CIPconnect for these applications and disconnect the module Ethernet port from the network

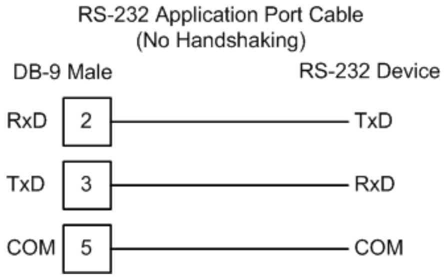
8.4 Application Port Cable Connection

The application ports on the MVI69E-GSC module support RS-232, RS-422, and RS-485 interfaces. Please inspect the module to ensure that the jumpers are set correctly to correspond with the type of interface you are using.

Note: When using RS-232 with radio modem applications, some radios or modems require hardware handshaking (control and monitoring of modem signal lines). Enable this in the configuration of the module by setting the UseCTS parameter to 1.

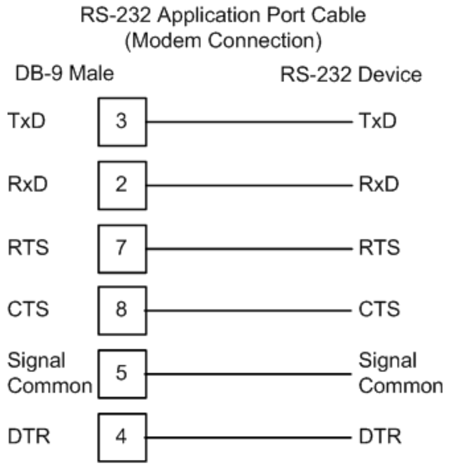
8.4.1 RS-232 Wiring

When the RS-232 interface is selected, the use of hardware handshaking (control and monitoring of modem signal lines) is user definable. If no hardware handshaking is used, here are the cable pin-outs to connect to the port.



RS-232: Modem Connection (Hardware Handshaking Required)

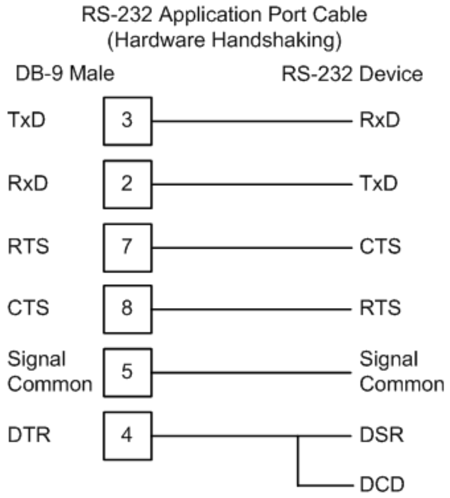
This type of connection is required between the module and a modem or other communication device.



The "Use CTS Line" parameter for the port configuration should be set to 'Y' for most modem applications.

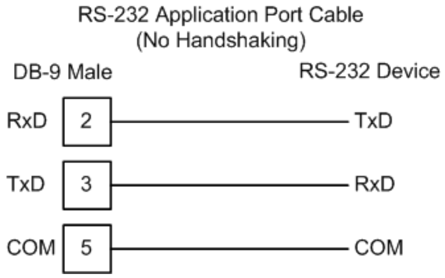
RS-232: Null Modem Connection (Hardware Handshaking)

This type of connection is used when the device connected to the module requires hardware handshaking (control and monitoring of modem signal lines).

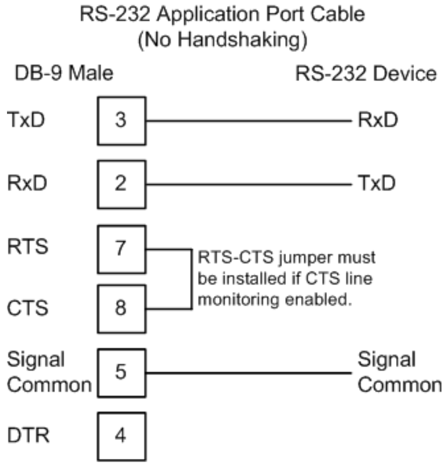


RS-232: Null Modem Connection (No Hardware Handshaking)

This type of connection can be used to connect the module to a computer or field device communication port.

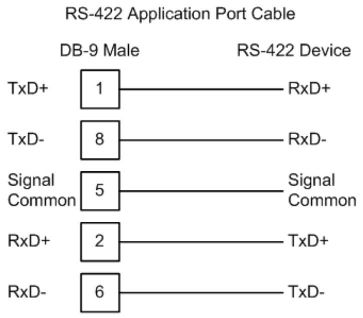


Note: For most null modem connections where hardware handshaking is not required, the *Use CTS Line* parameter should be set to **N** and no jumper is required between Pins 7 (RTS) and 8 (CTS) on the connector. If the port is configured with the *Use CTS Line* set to **Y**, then a jumper is required between the RTS and the CTS lines on the port connection.


RS-232 Application Port Cable
(No Handshaking)
DB-9 Male
RS-232 Device
TxD
3
RxD
RxD
2
TxD
RTS
7
RTS-CTS jumper must be installed if CTS line monitoring enabled.
CTS
8
Signal Common
5
Signal Common
DTR
4

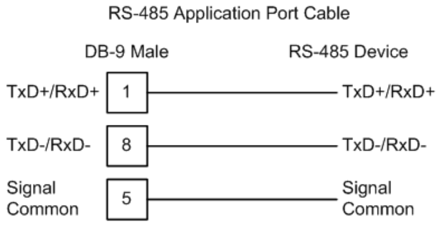
8.4.2 RS-422 Wiring

The RS-422 interface requires a single four or five wire cable. The Common connection is optional, depending on the RS-422 network devices used. The cable required for this interface is shown below:



8.4.3 RS-485 Wiring

The RS-485 interface requires a single two or three wire cable. The Common connection is optional, depending on the RS-485 network devices used. The cable required for this interface is shown below:



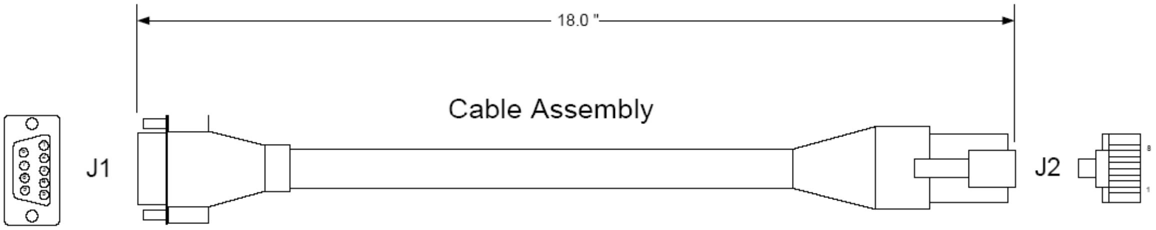
Note: This type of connection is commonly called a RS-485 half-duplex, 2-wire connection. If you have RS-485 4-wire, full-duplex devices, they can be connected to the gateway's serial ports by wiring together the TxD+ and RxD+ from the two pins of the full-duplex device to Pin 1 on the gateway and wiring together the TxD- and RxD- from the two pins of the full-duplex device to Pin 8 on the gateway. As an alternative, you could try setting the gateway to use the RS-422 interface and connect the full-duplex device according to the RS-422 wiring diagram. For additional assistance, please contact ProSoft Technical Support.

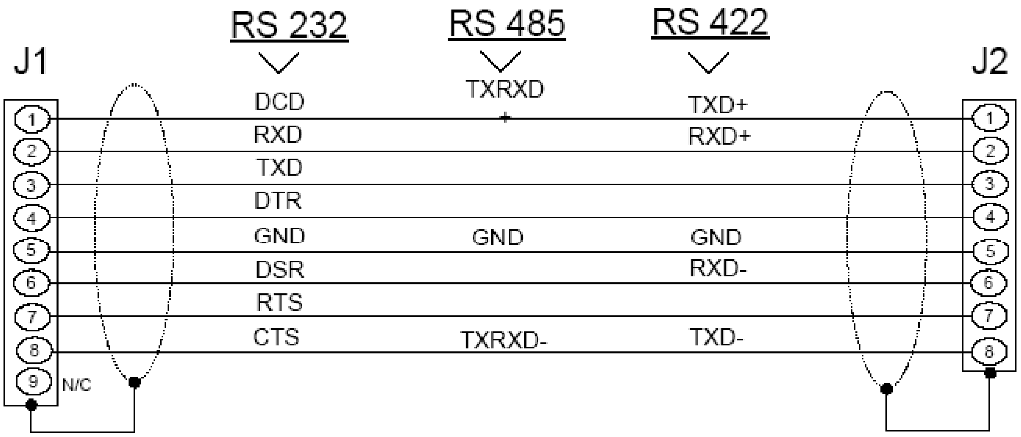
Note: Depending upon devices on the network, if there are problems in RS-485 communication that can be attributed to the signal echoes or reflections, then consider adding 120 OHM terminating resistors at both ends of the RS-485 line.

RS-485 and RS-422 Tip

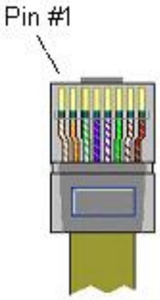
If communication in the RS-422 or RS-485 mode does not work at first, despite all attempts, try switching termination polarities. Some manufacturers interpret + and -, or A and B, polarities differently.

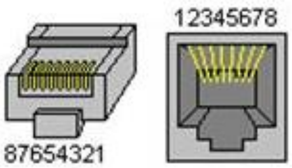
8.4.4 DB9 to RJ45 Adaptor (Cable 14)





Wiring Diagram





8.5 Status Data Definition

This section contains a description of the members present in the **GSCStatus** object. This data is transferred from the module to the processor as part of each read block.

Object in GSCInStat	Block Offset	Description
PassCnt	3	Program cycle counter
Product	4 to 5	Product name as ASCII string
Rev	6 to 7	Revision level as ASCII string
OP	8 to 9	Operating system level as ASCII string
Run	10 to 11	Run number as ASCII string
BlkErrs.Read	12	Number of blocks transferred from module to processor
BlkErrs.Write	13	Number of blocks transferred from processor to module
BlkErrs.Parse	14	Number of blocks parsed by module
BlkErrs.Err	15	Number of block errors in module
Port1.RxState	16	Port 1 receive state: -1 = Listening for data 1 = Receiving Port Data 2 = Waiting for Backplane transfer
Port1.RXCharCnt	17	Port 1 receive character count
Port1.RxMsgCnt	18	Port 1 receive block count
Port1.TxState	19	Port 1 transmit state: 0 = Waiting for Data to Send 1 = RTS On 2 = RTS Timeout 3 = Sending data 4 = Waiting for RTS Off 5 = RTS turned off 30 = Intercharacter Delay 31 = Intercharacter Delay 32 = Intercharacter Delay 100 = Message Delay before Transmit 101 = Message Delay before Transmit
Port1.TxCharCnt	20	Port 1 transmit character count
Port1.TxMsgCnt	21	Port 1 transmit block count
Port1.ErrorWord	22	Port 1 error word
Port2.RxState	30	Port 2 receive state: -1 = Listening for data 1 = Receiving Port Data 2 = Waiting for Backplane transfer
Port2.RXCharCnt	31	Port 2 receive character count
Port2.RxMsgCnt	32	Port 2 receive block count
Port2.TxState	33	Port 2 transmit state: 0 = Waiting for Data to Send 1 = RTS On 2 = RTS Timeout 3 = Sending data 4 = Waiting for RTS Off 5 = RTS turned off 30 = Intercharacter Delay 31 = Intercharacter Delay 32 = Intercharacter Delay 100 = Message Delay before Transmit 101 = Message Delay before Transmit

Object in GSCInStat	Block Offset	Description
Port2.TxCharCnt	34	Port 2 transmit character count
Port2.TxMsgCnt	35	Port 2 transmit block count
Port2.ErrorWord	36	Port 2 error word

GSCErrorWord Definition

Member Name	Bit in Word	Description
Cfg_type	Bit 0	The termination type configured for the port is not valid. Values between 0 and 15 are the only ones valid. The module will use type 0 (stream mode) for the port.
Cfg_Baud	Bit 1	The baud rate entered for the port is not valid. The module will use 9600 baud for the port.
Cfg_Parity	Bit 2	The parity value entered is not valid. Values between 0 and 4 are accepted. The module has set the parity to a value of none (0).
Cfg_DataBits	Bit 3	The number of data bits for the protocol is not valid. Values between 5 and 8 are accepted. The module assumes a value of 8 data bits.
Cfg_StopBits	Bit 4	The number of stop bits for the protocol is not valid. Values of 1 or 2 are accepted. The module assumes a value of 1 stop bit.
Cfg_Handshake	Bit 5	The handshake code for the port is not valid. The value entered must be in the range of 0 to 3. The module assumes a value of 0 (no handshaking).
Cfg_Rtermcount	Bit 6	The number of termination characters is not valid. The value must be set between 1 and 12 when using the termination character string to end a receive buffer. The module will not terminate a buffer when using the termination character(s) when this bit is set.
Cfg_RPacketLen	Bit 7	The number of characters for a packet is not valid. The value must be set between 1 and 4096 when the packet size termination option is used. The module will not use the packet length termination option when this bit is set.
Cfg_Rtimeout	Bit 8	The message timeout value is set to zero. The module will not use the message timeout termination option when this bit is set.
Cfg_Rdelay	Bit 9	The intercharacter delay value configured is set to zero. The module will not use the intercharacter delay option when this bit is set.
Cfg_Wtimeout	Bit 10	The write message timeout parameter is set to zero. The module assumes a value of 5000 milliseconds.
	Bit 11	
	Bit 12	
	Bit 13	
	Bit 14	
Err_ROverflow	Bit 15	Data is being received faster on the port than the ladder logic can process the read blocks. Alter the configuration of the module or the connected device. Receive data is being lost.

9 Support, Service & Warranty

9.1 Contacting Technical Support

ProSoft Technology, Inc. is committed to providing the most efficient and effective support ProSoft Technology, Inc. is committed to providing the most efficient and effective support possible. Before calling, please gather the following information to assist in expediting this process:

1. Product Version Number
2. System architecture
3. Network details

If the issue is hardware related, we will also need information regarding:

1. Module configuration and associated ladder files, if any
2. Module operation and any unusual behavior
3. Configuration/Debug status information
4. LED patterns
5. Details about the interfaced serial, Ethernet or Fieldbus devices

North America (Corporate Location)	**Europe / Middle East / Africa Regional Office**
Phone: +1 661-716-5100 ps.prosofttechnology@belden.com Languages spoken: English, Spanish REGIONAL TECH SUPPORT ps.support@belden.com	Phone: +33.(0)5.34.36.87.20 ps.europe@belden.com Languages spoken: English, French, Hindi, Italian REGIONAL TECH SUPPORT ps.support.emea@belden.com
Latin America Regional Office	**Asia Pacific Regional Office**
Phone: +52.222.264.1814 ps.latinam@belden.com Languages spoken: English, Spanish, Portuguese REGIONAL TECH SUPPORT ps.support.la@belden.com	Phone: +60.3.2247.1898 ps.asiapc@belden.com Languages spoken: Bahasa, Chinese, English, Hindi, Japanese, Korean, Malay REGIONAL TECH SUPPORT ps.support.ap@belden.com

For additional ProSoft Technology contacts in your area, please see:
www.prosoft-technology.com/About-Us/Contact-Us

9.2 Warranty Information

For details regarding ProSoft Technology's legal terms and conditions, please see:
www.prosoft-technology.com/ProSoft-Technology-Legal-Terms-and-Conditions

For Return Material Authorization information, please see:
www.prosoft-technology.com/Services-Support/Return-Material-Instructions